

IOWA STATE UNIVERSITY

Digital Repository

Graduate Theses and Dissertations

Iowa State University Capstones, Theses and
Dissertations

2010

Federated description logics for the semantic web

George Voutsadakis
Iowa State University

Follow this and additional works at: <https://lib.dr.iastate.edu/etd>

 Part of the [Computer Sciences Commons](#)

Recommended Citation

Voutsadakis, George, "Federated description logics for the semantic web" (2010). *Graduate Theses and Dissertations*. 11463.
<https://lib.dr.iastate.edu/etd/11463>

This Dissertation is brought to you for free and open access by the Iowa State University Capstones, Theses and Dissertations at Iowa State University Digital Repository. It has been accepted for inclusion in Graduate Theses and Dissertations by an authorized administrator of Iowa State University Digital Repository. For more information, please contact digirep@iastate.edu.

Federated description logics for the semantic web

by

George Voutsadakis

A dissertation submitted to the graduate faculty
in partial fulfillment of the requirements for the degree of
DOCTOR OF PHILOSOPHY

Major: Computer Science

Program of Study Committee:
Vasant Honavar, Co-major Professor
Giora Slutzki, Co-major Professor
Samik Basu
Jack Lutz
Jin Tian

Iowa State University

Ames, Iowa

2010

Copyright © George Voutsadakis, 2010. All rights reserved.

Ἐν οἶδα ὅτι οὐδέν οἶδα
Σωκράτης

Dum spiro spero
Marcus Tullius Cicero

This is the second Ph.D. thesis that I have written. A long time ago, I completed my first doctoral degree in Mathematical Logic, also at Iowa State, under the supervision of **Don Pigozzi**, one of the founding fathers of **Abstract Algebraic Logic**, and a student of **Alfred Tarski**.

To him I owe many of the things that I have learned about logic and life that have culminated in writing this dissertation. For better or for worse, it reflects who I am and the logical journeys that I continue to crave so badly in my life...

TABLE OF CONTENTS

LIST OF FIGURES	vi
ACKNOWLEDGEMENTS	vii
GENERAL INTRODUCTION	1
TECHNICAL INTRODUCTION	5
1. F-\mathcal{ALCI}: A FULLY CONTEXTUALIZED, FEDERATED LOGIC FOR THE SEMANTIC WEB	13
1.1 Introduction	13
1.2 The Federated Description Logic Language F- \mathcal{ALCI}	16
1.2.1 The Syntax	16
1.2.2 The Semantics	18
1.3 The Property of Exactness and a Characterization for F- \mathcal{ALCI}	21
1.4 A Reduction from F- \mathcal{ALCI} to \mathcal{ALCI}	24
1.5 Soundness of the Reduction	26
1.6 Completeness of the Reduction	33
1.7 Consequences of Soundness and Completeness	36
1.8 Summary	39
2. A FEDERATED TABLEAU ALGORITHM FOR F-\mathcal{ALCI}	40
2.1 Introduction	40
2.2 F- \mathcal{ALCI} Syntax and Semantics	43
2.3 Negation Local Form of Concept Expressions	45
2.4 Federated Tableaux for F- \mathcal{ALCI}	50

2.5	Tableau Algorithm for F- \mathcal{ALCI}	57
2.5.1	Federated Completion Graph	57
2.5.2	Federated Tableau Expansion	58
2.5.3	Synchronization	63
2.6	Correctness and Complexity	64
2.7	Summary and Discussion	74
3.	REASONING WITH F-\mathcal{ALCI} OVER LATTICES	76
3.1	Introduction	76
3.2	A Quick Review of \mathbf{L} - \mathcal{ALCI} and F- \mathcal{ALCI}	79
3.2.1	\mathbf{L} - \mathcal{ALCI} Syntax and Semantics	79
3.2.2	The Federated Language F- \mathcal{ALCI}	82
3.3	\mathbf{LF} - \mathcal{ALCI} : Reasoning with F- \mathcal{ALCI} over Lattices	84
3.4	A Reduction from \mathbf{LF} - \mathcal{ALCI} to \mathbf{L} - \mathcal{ALCI}	88
3.5	Soundness and Completeness of the Reduction \mathfrak{R}	89
3.5.1	Soundness	89
3.5.2	Completeness of the Reduction	94
3.5.3	Complexity	96
3.6	Summary	99
4.	PROBABILISTIC FEDERATED \mathcal{ALCI}	100
4.1	Introduction	100
4.2	A Quick Review of \mathcal{ALCI} and F- \mathcal{ALCI}	103
4.2.1	\mathcal{ALCI} Basics	103
4.2.2	F- \mathcal{ALCI} Basics	104
4.3	The Probabilistic Extension of F- \mathcal{ALCI}	106
4.3.1	Syntax	106
4.3.2	Semantics	109
4.4	A Modified P- \mathcal{ALCI}	118
4.4.1	Syntax and Semantics	119

4.4.2	P- \mathcal{ALCI} Problems of Interest	122
4.5	Reduction from Federated to Unimodule Problems	125
4.6	Soundness and Completeness	129
4.6.1	Soundness and Completeness for FPTCON	129
4.6.2	Soundness and Completeness for FTLEXENT	133
4.6.3	Algorithmic Significance and Complexities	136
4.7	Summary	137
5.	F-\mathcal{ALCIK}: FULLY CONTEXTUALIZED FEDERATED \mathcal{ALCI} WITH EPISTEMIC OPERATORS	139
5.1	Introduction	139
5.2	The Federated Languages F- \mathcal{ALCI} and F- \mathcal{ALCIK}	142
5.2.1	The Syntax	143
5.2.2	The Semantics	145
5.3	A Reduction from F- \mathcal{ALCIK} to \mathcal{ALCIK}	150
5.4	Soundness and Completeness of the Reduction \mathfrak{R}	154
5.5	Summary	162
	BIBLIOGRAPHY	164

LIST OF FIGURES

Figure 1.1	Interpretation of Contextualized Intersection.	19
Figure 1.2	Interpretation of Contextualized Negation.	20
Figure 1.3	A Non-exact Interpretation.	22
Figure 2.1	The distributed model of Example 4.	62
Figure 3.1	An Interpretation over $\mathbf{L}_{[0,1]}$	86
Figure 3.2	Another Interpretation over $\mathbf{L}_{[0,1]}$	87

ACKNOWLEDGEMENTS

I bear a huge scientific and personal debt to **Don Pigozzi**, my first advisor. His relentless pursuit of the scientific truth has ingrained in me the value of perseverance. His real sense of academic honesty, academic fairness and academic integrity I have pledged to abide by throughout my career.

The completion of this work would not have been possible without the guidance, the support and the advice of **Giora Slutzki**, **Vasant Honavar** and **Jie Bao**. A couple of years ago, Jie completed his doctoral dissertation under the guidance of Vasant and, together with Giora, they have contributed to the material included in several chapters of this thesis. I am very grateful for everything I have learned from them.

Many thanks are also due to the remaining members of my committee, **Jack Lutz**, **Samik Basu** and **Jin Tian**, for all their guidance and support. I have taken multiple classes with Jack, starting 15 years ago, and I have taken also classes with Samik and Jin. I am very grateful for all I have learned from them as well.

Finally, I would like to acknowledge partial support from the National Science Foundation, the Center for Information Protection at Iowa State University through a grant from the Boeing Corporation and the Department of Mathematics of Iowa State University.

GENERAL INTRODUCTION

This Ph.D. dissertation, referred to from now on as the (or this) dissertation (or this thesis), contains some contributions to the area of modular description logics for creating ontologies in the semantic web. This first section, titled General Introduction, contains a narrative and, mostly, non-technical introduction to the various topics studied in this dissertation. It does not contain formal definitions, nor does it provide precise statements or details of the results obtained. It does not include any references to preceding work on which this dissertation is based. It is solely intended to provide the reader with a general overview and an idea of the topics covered. Technical details and references, which increase precision, will be postponed to the technical introduction that follows and to the introductions of the various individual chapters of the dissertation.

The dissertation consists of five chapters, all of which contain contributions to the general area of knowledge representation in artificial intelligence. Virtually all logic-based applications in artificial intelligence require methods for representing the knowledge of the participating agents, for retrieving and reasoning with that knowledge and for making decisions based on that knowledge. This has become recently increasingly important in applications involving information that is stored in the world-wide web. To organize information stored in the web and to logically manipulate such information, it is required that it be represented in such a way so that its intended semantics is well-defined and unambiguous. One way of achieving this is by representing information using semantic, or ontology, languages and using accompanying automated reasoners that support drawing logical inferences. Most of the formal languages that have been developed for this task are based on decidable fragments of first-order logic or decidable extensions of such fragments with additional features, such as modal operators

or probabilistic constructors. These are collectively known as description logics. They have been at the forefront of research in knowledge representation for several decades and there is a well-established workshop, the annual International Workshop on Description Logics, where new ideas, advances and results in this area are presented.

The recent dominance of the web in several important human activity domains, such as science, communications, commerce, government etc., and the distributed nature of the world-wide web has encouraged the diversification of the area of knowledge representation, in general, and of description logics, in particular, towards studying paradigms for representing, retrieving and reasoning with knowledge that is repositied by various independent individuals or organizations in various physical locations in the web. However, the various parts constituting this collective knowledge are not entirely disjoint. For instance, many web sites have links or references to other web sites, where better or more extensive explanations of the terms that they are using may be found. Therefore, a semantics that makes these interconnections precise and coordinates reasoning with pieces of information originating from various modules is necessary. In description logics, this new shift in emphasis has led to the introduction and study of modular description logics. These are logical systems that consist of independently developed modules over the same logical language, but each with different local non-logical symbols, such as constants, concept names and role names, which represent, respectively, elements of the domain of discourse, sets of elements and relations between elements. Apart from the general workshop on description logics, mentioned above, there is, by now, a well-established more specialized workshop, the annual International Workshop on Modular Ontologies, which focuses in both applied and theoretical aspects of knowledge representation and reasoning in distributed environments and, more specifically, in methods for organizing and reasoning with information on the semantic web.

In the dissertation various contributions are made to the area of modular description logics. More specifically, in Chapter 1, we study a novel modular description logic, called $F\text{-}\mathcal{ALCI}$, or Federated \mathcal{ALCI} . It is a modular extension of the well-known basic description logic \mathcal{ALCI} , incorporating features, like modularity and full contextualization of meanings for the logical

connectives, that have not been studied together before. In Chapter 2, a distributed reasoning algorithm is developed for processing and drawing conclusions from the information that is stored in various modules of a federated ontology using the language $F\text{-}\mathcal{ALCI}$. The theoretical time complexity of the algorithm is not better than that of a centralized algorithm applied on a single ontology resulting from integrating the various modules in a context-sensitive way. Having, however, a distributed algorithm has the great advantage of making integration unnecessary. On many occasions integrating the various modules in a single ontology may be impossible due to the sheer sizes of the modules or due to the reluctance of the owners of the modules to allow remote use of their entire local ontologies. In Chapter 3, we introduce an extension of $F\text{-}\mathcal{ALCI}$ that is able to handle imprecise or vague knowledge. In many applications, such as commerce, security etc., handling imprecise knowledge is of paramount importance. For instance, the concept **Expensive** when referring to commodities or the concept **Secure** when referring to communications or classified information might not be crisply defined. An item might be moderately expensive or very expensive or a piece of information may be classified or top secret. Thus, membership of individuals of a domain in a given set, or pairs of individuals in a given relation, may only be true or false to a specific degree of certainty. The language $LF\text{-}\mathcal{ALCI}$ has the same syntax as $F\text{-}\mathcal{ALCI}$, but its semantics allows representations of memberships in concepts and roles that are not either true or false, but are, rather, chosen from elements of a certainty lattice \mathbf{L} , which specify the degrees to which these membership relations are believed to be valid. The language $LF\text{-}\mathcal{ALCI}$ is the first language that combines the features of modularity and of reasoning with imprecise information. In Chapter 4, we study another extension of $F\text{-}\mathcal{ALCI}$, one that is able to incorporate probabilistic statements. For instance, in a knowledge base consisting of information regarding the academic personnel at a college campus, we might want to express the fact that, although not all lecturers are fully insured, they are fully insured with probability at least 0.7. In the probabilistic extension $PF\text{-}\mathcal{ALCI}$ of $F\text{-}\mathcal{ALCI}$, presented in Chapter 4, we allow statements of the form $(\text{FullyInsured}|\text{Lecturer})[0.7, 1]$, which capture exactly this intuition. Again, we believe that $PF\text{-}\mathcal{ALCI}$ is the first description logic that combines modularity with probabilistic features.

Finally, in the last chapter, Chapter 5, we study the language $F\text{-}\mathcal{ALC}\mathcal{IK}$, yet another extension of $F\text{-}\mathcal{ALC}\mathcal{I}$, which includes epistemic operators that, effectively, transform the underlying language $\mathcal{ALC}\mathcal{I}$ to $\mathcal{ALC}\mathcal{IK}$, which is a fragment of first-order non-monotonic modal logic. In the single module case, the introduction of the epistemic operator formalizes procedural rules and, in addition, enables more sophisticated forms of query formulation. Among these are various forms of closed-world reasoning. It enables the queries to refer both to aspects of the external world, as represented by the knowledge base, and to aspects of what the knowledge base knows about the external world. These features are inherited also by the modular language $F\text{-}\mathcal{ALC}\mathcal{IK}$.

TECHNICAL INTRODUCTION

This dissertation consists of five chapters, all of which contain contributions to the area of modular description logics, a subarea of the area of knowledge representation in Artificial Intelligence.

The current form of the world-wide web consists mostly of information geared towards human understanding and consumption. Because of the large amount of data and services that have, and continue to be, accumulated over the last few years, it has become increasingly important to be able to represent them in a way that would make them amenable to machine analysis and processing. This, in turn, implies that the duality between the stored data and the information they represent should be taken into account in earnest and formally established. As a result, logic-based languages, with formally defined semantics, have been proposed as the appropriate languages for use when representing information in the web. This latter vision of the web has been termed the *semantic web* [11].

In the semantic web, at its most general abstract level, information is organized in groups, each of which consists of information about a specific domain of discourse, e.g., pharmacology, veterinary medicine, food industry, wine-making, etc. Such a group is called an *ontology*. Special logic-based knowledge representation languages have been devised to construct ontologies. The one currently recommended by the W3C Consortium is the OWL ontology language [50, 9]. Ontology languages are based on underlying logical languages which are, most often, decidable fragments of first-order logic. These have been called *description logics* [1]. In general, a description logic allows one to define *concepts*, which are unary relations over the domain of discourse, and *roles*, which are binary relations over the domain of discourse. Hierarchies of concepts and of roles based on inclusion relationships compose the *TBox* (terminological

box) of an ontology defined using a description logic. On the other hand, a description logic allows also the representation of individuals in the domain of discourse using *constants*, and memberships of these individuals in a concept or of pairs of these individuals in roles compose the *ABox* (assertional box) of the ontology. Various description logics have been proposed over the years. They differ depending on the constructors that they allow in the definition of new concepts and roles based on existing ones. A general tradeoff in this area is between the *expressiveness* of a description logic and the *computational complexity* of deciding problems, such as concept satisfiability, for expressions in the description logic [58].

Another special feature of the information that has dominated the development of the world-wide web is the fact that it is contributed by various independent individuals or organizations and repositied in various physical locations in the web. Typically, the modules developed in this way cover each a distinct domain of expertise. But the various modules include data referring to partially overlapping information. For instance, a module developed by wine experts may need to reuse some terminology or other information included in a module developed by culinary experts in order to recommend combinations of various wines with matching meals. This interdependence is manifested in the web in various ways, for instance, by using links to other web-sites or by referring to them for further information and clarification of borrowed concepts and definitions.

In the envisioned semantic web, on the other hand, the combined need for independent but interactive development and for transparent semantics has led to the consideration of *modular ontologies*. The main goal here is, not only to be able to borrow or refer to data included in other, independently created, web-sites, but to also have a clear meaning of how this borrowed data is supposed to be interpreted in the context of the newly created ontology and the information that it represents. To cope with this need, ontology languages have to be extended to support modularity and contextuality [68]. That is, they should provide means for developing various modules independently, partially importing information from other, already existing or simultaneously developed, modules and for interpreting the imported information in the new context where it is used, in connection with both the local information and the

remote source where it had been originally defined.

At the level of description logics, this modularity requirement is shifting focus from ordinary description logics to *modular description logics*. These are logical languages, based on ordinary description logics that, roughly speaking, provide the additional support required for modularization. In other words, they allow creation of various modules, based on the same logical language, but make special provision for importing foreign terminology and information and for providing specialized semantics to clarify how the imported information is to be interpreted locally and how it should interact with locally defined information. There have various proposals for modular description logics. They differ depending on a variety of characteristics. One is the *underlying description logic* used in each module to express local information. Another is the collection of constructs employed to allow *importing* foreign concepts and/or roles in the newly created module. Yet another is the *semantics* used to interpret imported data and, in particular, how context sensitive this interpretation is intended to be.

Examples of proposed modular description logics include distributed description logics (DDL) [13, 40], \mathcal{E} -Connections [28], semantic importing [73], semantic binding [104], and package-based description logics (P-DL) [5, 8]. An alternative approach to importing knowledge, or, rather, reusing knowledge from existing ontologies, relies on the logical notion of conservative extension [41, 24, 23].

In Chapter 1 of the thesis, we study a novel modular description logic, called F- \mathcal{ALCI} , or Federated \mathcal{ALCI} . It is a modular extension of the well-known basic description logic \mathcal{ALCI} [85] (see, also, [2]), incorporating features, like modularity and full contextualization of meanings for the logical connectives, that have not been studied together before. More precisely, the language has the following desirable features:

- Provision for use of the relatively rich description logic (DL) \mathcal{ALCI} within each ontology module.
- Contextualized interpretation of each logical connective used within the DL modules.
- Guarantee that the results of reasoning are always the same as those obtained by a

standard reasoner over an integrated ontology resulting from combining the relevant parts of the individual modules in a context-specific manner.

- Relaxation of the severe restrictions on the semantics imposed, in particular, on the Package-based Description Logics (P-DL) paradigm.

On the other hand, some drawbacks, as compared to other approaches are:

- Other modular approaches have been studied based on even richer DLs than \mathcal{ALCT} .
- We relegate the study of ABoxes to future work unlike other approaches which have already dealt with ABoxes.
- The relaxation of the P-DL semantics occurs at the expense of losing some desirable properties, such as propagation of subsumption and preservation of unsatisfiability, but it is shown how these may be recovered by tightening the semantics, without, however, making it as restrictive as that of P-DL.

Among the most important decision problems for DLs and, by extension, for modular DLs is the *concept satisfiability problem*. This asks whether, given a TBox and a concept expression in the description logic under consideration, there exists a model of the TBox in which the extension of the concept expression is a nonempty subset of the domain. The importance of this problem lies, in part, to the fact that, for relatively expressive DLs, such as \mathcal{ALCT} , many other decision problems of interest are reducible to concept satisfiability (see, e.g., [1, 95]). A common method for showing the decidability of this problem in modular description logics is to devise, if possible, a sound and complete reduction of the problem into the corresponding problem for a similar ordinary (unimodule) description logic, in which concept satisfiability has already been shown to be decidable. Such a reduction has been performed both for DDL in [14] and for P-DL [8]. If the concept satisfiability problem for the ordinary DL is of known computational complexity, then the reduction often allows one to pinpoint the complexity of the same problem for the corresponding modular DL. In Chapter 1 a reduction from F- \mathcal{ALCT} to \mathcal{ALCT} is given and this allows, as described above, the derivation of both the decidability and

the computational complexity of this problem for $F\text{-}\mathcal{ALCT}$, taking advantage of corresponding results for \mathcal{ALCT} .

Even though sound and complete reductions of a modular description logic to a corresponding ordinary description logic show that, in principle, reasoning in the modular setting may be simulated in the single-module framework, this counteracts the very reason for creating and developing modular languages. Their nature and *raison d'être* in supporting independent ontology development, while allowing interactions between the modules, requires the development of “genuine” distributed algorithms. These algorithms should allow reasoning in a cooperative fashion between the local reasoners at the various physical locations where the distinct modules are residing, without the need to first transfer all modules into a single location and to integrate them into a single ontology.

Numerous researchers, including Serafini et al. [81, 83, 82], have argued for such a genuinely modular approach and have designed specific systems/architectures (DRAGO) to facilitate decentralized reasoning over distributed ontologies. Some authors advocated partitioning of large ontologies into smaller but computationally more efficient ontologies [94, 80]. In fact, several algorithms have been presented for the three paradigms mentioned above. Serafini et al. [81, 83, 82] introduced a tableau algorithm for reasoning with DDL. Grau et al. [28, 27] present a tableau procedure for \mathcal{E} -Connections. Finally, Bao et al. [4] present distributed reasoning algorithms for P-DLs.

In Chapter 2 of the thesis, a *distributed reasoning algorithm* is developed for processing and drawing conclusions from the information that is stored in various modules of a federated ontology that uses the language $F\text{-}\mathcal{ALCT}$. Although the time requirements of the algorithm are not better than those of a centralized algorithm applied on a single ontology, resulting from integrating the various modules in a context-sensitive way, the given algorithm has the advantage of making integration unnecessary.

Much work in the area of ontology languages and description logics has been devoted to extending ordinary DLs by adding features that would enable users to reason with imprecise or uncertain knowledge and, also, perform epistemic reasoning. However, not many attempts

have been carried out in combining these features with modularity so as to obtain a modular description logic that can accommodate uncertain or epistemic reasoning. The three remaining chapters of this first part of the dissertation deal with such extensions of the basis modular DL $F\text{-}\mathcal{ALCI}$, which is the subject of Chapters 1 and 2.

In Chapter 3, an extension $LF\text{-}\mathcal{ALCI}$ of $F\text{-}\mathcal{ALCI}$ is presented that maintains the same syntax as the original language, but is geared towards a semantics over an arbitrary complete, completely distributive lattice \mathbf{L} with a negation operation, rather than over the usual 2-element Boolean lattice. The elements of the lattice \mathbf{L} are supposed to represent certainty values. Accordingly, the lattice \mathbf{L} is dubbed a *certainty lattice*. Membership of an individual element of the domain of discourse in the extension of a concept, or of a pair of individuals in the extension of a role, is assigned a certainty value from \mathbf{L} , rather than being either true or false. This value represents the degree of confidence in the corresponding membership relation. This framework allows both a quantitative kind of reasoning, e.g., when taking the continuous $[0, 1]$ real interval as the underlying lattice, as well as a qualitative kind, when, for instance, one considers a four element lattice, with values true, false, uncertain and inconsistent.

Knowledge management of uncertain or imprecise information has been considered before, for instance in [47, 53, 64] using probability theory, [49] using possibility theory, [75, 90] using many-valued logics and [12, 48, 91, 92, 96, 103] using fuzzy logic. This list is not exhaustive.

In the unimodule setting the ordinary description logic \mathcal{ALC} has been endowed with a lattice semantics by Straccia [93]. We follow his work and use his results to show that our modular logic is decidable and to pinpoint the complexity of the concept satisfiability problem. Our work is in some respects more general: It deals with the modular setting, adds to the underlying DL inverse roles and introduces full contextualization of all logical connectives. On the other hand, Straccia's work is more general in that it allows reasoning with both TBoxes and ABoxes. We restrict ourselves only to TBoxes, for the sake of simplicity, since we are making a first attempt at building a modular DL combining all the aforementioned features.

An alternative approach in dealing with uncertain or imprecise information to reasoning over an arbitrary uncertainty lattice is to use probabilistic methods. This has been explored

previously in various areas in artificial intelligence, such as logic programming [62], knowledge representation [47, 53, 54, 55, 64] and ontology engineering [33, 34, 102, 77, 37, 72, 30, 31]. In Chapter 4 of the dissertation, based on previous work of Lukasiewicz [64], a modular description logic $\text{PF-}\mathcal{ALCT}$, extending $\text{F-}\mathcal{ALCT}$ with probabilistic features, is introduced.

More precisely, probabilistic terminological axioms will be added to the language $\text{F-}\mathcal{ALCT}$ to obtain the probabilistic federated description logic $\text{PF-}\mathcal{ALCT}$. Lukasiewicz in [64] introduces probabilistic analogs of the very expressive description logics $\mathcal{SHIF}(\mathbf{D})$ and $\mathcal{SHOIN}(\mathbf{D})$. Again, since our effort constitutes a first attempt at the creation of a relatively expressive modular description logic with probabilistic features, we opt to deal with a rather simplified version of the description logic used in [64]. In particular, when compared with the underlying language used by Lukasiewicz, our language has the following three limitations:

- It uses the significantly less expressive \mathcal{ALCT} as the DL inside each module.
- Second, it does not allow the use of concrete domains, as does Lukasiewicz’s treatment.
- Finally, we deal only with terminological axioms, whereas Lukasiewicz deals with both terminological and assertional axioms.

However, our work again constitutes a first attempt at combining several features collectively for the first time in an ontology language: modularity, context and probabilistic features. Moreover, we show that several of the nice probabilistic features of Lukasiewicz’s approach pertaining to default and probabilistic terminological axioms still hold in the distributed context, despite the limited expressivity of the underlying description logic.

Besides reasoning with uncertainty, non-monotonic reasoning and reasoning with procedural rules are features that one may want to introduce to a description logic, but are not directly expressible in fragments of first order logic. To accommodate these features in DLs, Donini et al. introduced in [35] an epistemic operator in the description logic \mathcal{ALC} [85]. This operator effectively extends \mathcal{ALC} to a fragment of first-order non-monotonic modal logic. It achieves the goal of formalizing procedural rules and allows more sophisticated forms of query formulation, including various forms of closed-world reasoning. The knowledge operator in [35]

is interpreted as in [79, 59]. This enables the queries to refer both to aspects of the external world, as represented by the knowledge base, and to the epistemological aspect of what the knowledge base knows about the external world.

In the last chapter, Chapter 5, of this thesis the federated description logic $F\text{-}\mathcal{ALCI}$ is extended to accommodate these epistemic features. More precisely, epistemic operators are incorporated into $F\text{-}\mathcal{ALCI}$, thus, creating a new modular ontology language $F\text{-}\mathcal{ALCIK}$, that allows exploiting non-monotonic aspects of reasoning and the expression of procedural rules. This is, to the best of our knowledge, the first language that combines modularity, contextualization and epistemic operators. When only one module is present, $F\text{-}\mathcal{ALCIK}$ reduces to the ordinary DL \mathcal{ALCIK} , an extension with inverse roles of the language \mathcal{ALCK} , studied in [35].

For all extensions of the federated language $F\text{-}\mathcal{ALCI}$, that are presented in Chapters 3,4 and 5 of the dissertation, sound and complete reductions are provided to corresponding uni-module languages. These are exploited in each chapter to prove decidability of the newly introduced modular ontology language. Furthermore, relying on known results on the computational complexity of the concept satisfiability problem in the single-module case, we obtain estimates of the computational complexity of the same problem in the modular case. Although, we do not present in detail algorithms for solving concept satisfiability for the three extensions of $F\text{-}\mathcal{ALCI}$, $LF\text{-}\mathcal{ALCI}$, $PF\text{-}\mathcal{ALCI}$ and $F\text{-}\mathcal{ALCIK}$, treated in Chapters 3,4 and 5, respectively, we, nevertheless, indicate how the algorithms available in the literature for the uni-module languages may be modified and combined with our reductions to provide sound and complete decision procedures for their modular counterparts.

In general, we postpone dealing with ABoxes and constructing genuinely distributed algorithms for these problems, such as the one devised for $F\text{-}\mathcal{ALCI}$ itself in Chapter 2, to future research.

Chapter 1. F- \mathcal{ALCI} : A FULLY CONTEXTUALIZED, FEDERATED LOGIC FOR THE SEMANTIC WEB

Abstract

We introduce F- \mathcal{ALCI} , a federated version of the description logic \mathcal{ALCI} . An F- \mathcal{ALCI} ontology, like its *package-based* counterpart \mathcal{ALCIP}^- , consists of multiple \mathcal{ALCI} ontologies that can *import* concepts or roles defined in other modules. Unlike \mathcal{ALCIP}^- , which supports only contextualized negation, F- \mathcal{ALCI} , supports contextualization of each of the logical connectives, a feature that allows more flexible reuse of knowledge from independently developed ontologies. We provide a new semantics for F- \mathcal{ALCI} based on image domain relations and establish the conditions that need to be imposed on domain relations to ensure properties, such as preservation of unsatisfiability and monotonicity of inference, that are desirable in distributed web applications. We also establish the *decidability* of F- \mathcal{ALCI} .

1.1 Introduction

Recent efforts aimed at enriching the world-wide web with machine interpretable content and interoperable resources and services, are transforming the web into the *semantic web* [11]. The semantic web, much like the world-wide web, relies on the *network effect*, that is, on leveraging the work of independent actors who contribute resources that are interlinked to form a web of resources. The ontologies that provide a basis for establishing the intended semantics of resources that constitute the semantic web (databases, knowledge bases, services) are typically developed independently to serve the needs of specific communities. They typically cover different, partially overlapping, domains of discourse (e.g., biology, medicine, pharmacology).

Inevitably, the axioms that make up the ontologies are applicable within the *contexts* that are implicitly assumed by their authors. However, many application scenarios require selective use of knowledge from multiple independently developed ontology modules. For example, a group that is focused on translating discoveries that link genetic and environmental factors to specific diseases into effective therapies might need to selectively reuse the contents of an ontology created for use in one context (e.g., genetic studies) in a different, but related context (e.g., drug design). Reaping the benefits of the network effect in such a setting requires theoretically well-founded yet practically useful approaches to selective, context-sensitive reuse of knowledge from autonomous, distributed, ontology modules.

Early recognition of the importance of careful treatment of context in artificial intelligence systems [68] was followed by work on non-monotonic reasoning [68, 67, 70, 78], propositional and quantificational (first order) logics of context [22, 20, 19, 21] and context-based logics for distributed knowledge representation and reasoning [42, 39, 38, 15]. More closely related to contextualizing information in the semantic web are the references [44, 74, 88, 89, 10].

Recently, several modular ontology formalisms that support reuse of knowledge from multiple distributed ontology modules, have been explored. Examples include distributed description logics (DDL)[13, 40], \mathcal{E} -Connections [28], semantic importing [73], semantic binding [104], and package-based description logics (P-DL) [5, 8]. Such frameworks typically assume that the individual ontology modules are expressed in some decidable family of description logics (DL) and provide constructs for the sharing of knowledge across ontology modules. An alternative approach to knowledge reuse relies on a particular notion of modularity of knowledge bases based on the notion of *conservative extensions* [41, 24, 23], which allows ontology modules to be interpreted using standard semantics by requiring that they share the same interpretation domain.

Existing modular ontology formalisms offer only limited ways to connect ontology modules, and hence, limited ability to reuse knowledge across modules. For instance, DDL does not allow concept construction using foreign roles or concepts, or guarantee the transitivity of inter-module concept subsumptions, known as *bridge rules*. It has recently been shown

that allowing negated roles or cardinality restrictions in bridge rules or inverse bridge rules, where the bridge rules are used to connect \mathcal{ALC} -ontologies, makes the resulting DDL ontology undecidable [7]. The \mathcal{E} -Connections formalism does not allow concept subsumptions across ontology modules or the use of foreign roles. Conservative extensions [24, 23, 65] require a single global interpretation domain. Thus, the designers of different ontology modules have to anticipate all possible contexts in which knowledge from a specific module might be reused thereby precluding flexible and selective reuse of knowledge across ontology modules.

P-DL offers a richer syntax than the previous approaches but, to preserve contextuality of knowledge and transitivity of role inclusions across ontology modules, and to guarantee decidability of the resulting logic, the Package-based DL \mathcal{SHOIQP} [8] imposes several restrictions. In particular, P-DL requires partial isomorphisms between various local domains that are related via domain relations (that is, functions that link individuals across the different local domains). Thus, it is interesting to explore whether some of these conditions can be relaxed, in the process simplifying the semantics as well as the design of federated reasoners for the resulting logic.

This paper, on the other hand, explores a family of logics, which we call *contextualized federated description logics* (CFDLs), paying special attention to characterization of the tradeoffs between specific restrictions on semantics and some of the desirable features that are offered by P-DLs. Specifically, we focus on the language $\text{F-}\mathcal{ALCI}$, which is the contextualized federated counterpart of the P-DL \mathcal{ALCIP}^- , and where each of the individual ontology modules is expressed in the DL \mathcal{ALCI} . Some features of CFDL $\text{F-}\mathcal{ALCI}$ include:

- Provision for use of a relatively rich DL within each ontology module.
- Contextualized interpretation of each logical connective used within the DL modules (unlike in P-DLs [5, 8] where only negation is contextualized). Locality of axioms in ontology modules is obtained “for free” by its *contextualized semantics*. Thus, as in the case of P-DLs, inferences are always drawn *from the point of view* of a *witness* module. It follows that different modules might infer different consequences, based on the knowledge that they import from other modules.

- Guarantee that the results of reasoning are always the same as those obtained by a standard reasoner over an integrated ontology resulting from combining the relevant parts of the individual ontologies in a context-specific manner.
- Relaxation of the severe restrictions imposed on the P-DL semantics as much as possible while, at the same time, retaining the desirable properties of P-DLs.

In particular, we show that in the general case, when only the most relaxed restrictions are imposed on the semantics of $F\text{-}\mathcal{ALCI}$, many of the properties that one might want to satisfy, like, e.g., monotonicity of inference and preservation of unsatisfiability, are lost. Regaining these properties requires strengthening the conditions on the semantics of $F\text{-}\mathcal{ALCI}$. Thus, the major contribution of this paper is a characterization of the tradeoffs between restrictions on semantics and some of the desirable features of P-DLs. Specifically, we show that it is possible to preserve many of the desirable properties of P-DLs, while at the same time imposing milder restrictions.

1.2 The Federated Description Logic Language $F\text{-}\mathcal{ALCI}$

In [8], given an ordinary description logic \mathcal{L} , the notation \mathcal{LP} is introduced to denote its package-based counterpart, i.e., the package-based description logic which uses \mathcal{L} as the logical language in each of its packages. Furthermore, the notation \mathcal{LP}^- signifies that the importing of concept names and role names across packages is acyclic. In the present work, we use the prefix “F-”, standing for **F**ederated, to denote a contextualized federated language and, since our discussion is limited to acyclic importing, we omit the use of a superscript “-” from the notation.

In this section, the syntax and the semantics of the language $F\text{-}\mathcal{ALCI}$ will be described in some detail.

1.2.1 The Syntax

Suppose a directed acyclic graph $G = \langle V, E \rangle$, with $V = \{1, 2, \dots, n\}$, is given. The intuition is that its n nodes correspond to local modules of a modular ontology and its edges correspond

to the importing relations between these modules. For technical reasons, we add a loop on each vertex of G .

For every node $i \in V$, the signature of the i -language always includes a set \mathcal{C}_i of **i -concept names** and a set \mathcal{R}_i of **i -role names**. We assume that all sets of names are pairwise disjoint. Out of these, a set of *i -concept expressions* $\widehat{\mathcal{C}}_i$ and a set of *i -role expressions* $\widehat{\mathcal{R}}_i$ are built.

Recall that the description logic \mathcal{ALCI} allows concept expressions that are constructed recursively from its signature symbols, i.e., its role and concept names, using negation, conjunction, disjunction, value and existential restriction and inverses of role names. Its formulas are subsumptions between concept expressions.

The syntax of the description logic $F\text{-}\mathcal{ALCI}$ is defined as follows:

Definition 1 (Roles and Concepts) *The set of i -roles or i -role expressions $\widehat{\mathcal{R}}_i$ consists of expressions of the form R, R^- , with $R \in \mathcal{R}_j, (j, i) \in E$.*

The set of i -concepts or i -concept expressions $\widehat{\mathcal{C}}_i$, on the other hand, is defined recursively as follows:

$$A \in \mathcal{C}_j, \top_j, \perp_j, \neg_j C, C \sqcap_j D, C \sqcup_j D, \exists_j R.C, \forall_j R.C, \quad (1.1)$$

where $(j, i) \in E$, $C, D \in \widehat{\mathcal{C}}_i \cap \widehat{\mathcal{C}}_j$ and $R \in \widehat{\mathcal{R}}_i \cap \widehat{\mathcal{R}}_j$.

Using the concepts and roles of $F\text{-}\mathcal{ALCI}$, we define its formulas, as follows:

Definition 2 (Formulas) *The i -formulas are expressions of the form $C \sqsubseteq D$, with $C, D \in \widehat{\mathcal{C}}_i$, for all $i \in V$.*

An $F\text{-}\mathcal{ALCI}$ -**TBox** or **TBox** is a collection $T = \{T_i\}_{i \in V}$, where T_i is a finite set of i -formulas, for all $i \in V$, called the *i -TBox*. Since, in this paper, we do not consider RBoxes or ABoxes, the terms *TBox*, *ontology* and *knowledge base* will be used interchangeably.

We use, for every $i \in V$, the notation $\overline{\mathcal{R}}_i$ and $\overline{\mathcal{C}}_i$ to denote the set of i -roles and of i -concepts, respectively, that occur in T_i . $\overline{\mathcal{C}}_i$ is a finite subset of $\widehat{\mathcal{C}}_i$, for every $i \in V$. A role name in $\mathcal{R}_j \cap \overline{\mathcal{R}}_i$ or a concept name in $\mathcal{C}_j \cap \overline{\mathcal{C}}_i$ is said to be **imported from** module j **to** module i . Furthermore, since $\overline{\mathcal{C}}_i \subseteq \widehat{\mathcal{C}}_i$, it is obvious that a module i is allowed to use logical connectives subscripted by the index of a module j , whenever $(j, i) \in E$.

1.2.2 The Semantics

In this subsection, we present the semantics for the language $\text{F-}\mathcal{ALCT}$.

Definition 3 *An interpretation $\mathcal{I} = \langle \{\mathcal{I}_i\}_{i \in V}, \{r_{ij}\}_{(i,j) \in E} \rangle$ consists of a family $\mathcal{I}_i = \langle \Delta^i, \cdot^i \rangle$, $i \in V$, of **local interpretations**, together with a family of **image domain relations** $r_{ij} \subseteq \Delta^i \times \Delta^j$, $(i, j) \in E$, such that $r_{ii} = \text{id}_{\Delta^i}$, for all $i \in V$.*

Notation: For a binary relation $r \subseteq \Delta^i \times \Delta^j$, $X \subseteq \Delta^i$ and $S \subseteq \Delta^i \times \Delta^i$, we set

$$r(X) := \{y \in \Delta^j : (\exists x \in X)((x, y) \in r)\},$$

$$r(S) := \{(z, w) \in \Delta^j \times \Delta^j : (\exists (x, y) \in S)((x, z), (y, w) \in r)\}$$

to denote the images of X and S under the binary relation r .

A local interpretation function \cdot^i interprets i -role names and i -concept names, as well as \perp_i and \top_i , as follows:

- $C^i \subseteq \Delta^i$, for all $C \in \mathcal{C}_i$,
- $R^i \subseteq \Delta^i \times \Delta^i$, for all $R \in \mathcal{R}_i$,
- $\top_i^i = \Delta^i$, $\perp_i^i = \emptyset$.

The interpretations of imported role names and imported concept names are computed by the following rules:

- $C^i = r_{ji}(C^j)$, for all $C \in \mathcal{C}_j \cap \widehat{\mathcal{C}}_i$,
- $R^i = r_{ji}(R^j)$, for all $R \in \mathcal{R}_j \cap \widehat{\mathcal{R}}_i$,
- $\top_j^i = r_{ji}(\Delta^j)$, $\perp_j^i = \emptyset$.

The recursive features of the local interpretation function \cdot^i are as follows:

- $R^{-i} = R^{i-}$, for all $R \in \mathcal{R}_i$,

- $(\neg_j C)^i = r_{ji}(\Delta^j - C^j)$
- $(C \sqcap_j D)^i = r_{ji}(C^j \cap D^j)$
- $(C \sqcup_j D)^i = r_{ji}(C^j \cup D^j)$
- $(\exists_j R.C)^i = r_{ji}(\{x \in \Delta^j : (\exists y)((x, y) \in R^j \text{ and } y \in C^j)\})$
- $(\forall_j R.C)^i = r_{ji}(\{x \in \Delta^j : (\forall y)((x, y) \in R^j \text{ implies } y \in C^j)\})$

For all $i \in V$, i -**satisfiability**, denoted by \models_i , is defined by $\mathcal{I} \models_i C \sqsubseteq D$ iff $C^i \subseteq D^i$. Given a TBox $T = \{T_i\}_{i \in V}$, the interpretation \mathcal{I} is a **model** of T_i , written $\mathcal{I} \models_i T_i$, iff $\mathcal{I} \models_i \tau$, for every $\tau \in T_i$. Moreover, \mathcal{I} is a **model** of T , written $\mathcal{I} \models T$, iff $\mathcal{I} \models_i T_i$, for every $i \in V$.

Let $w \in V$. Define $G_w = \langle V_w, E_w \rangle$ to be the subgraph of G induced by those vertices in G from which w is reachable and $T_w^* := \{T_i\}_{i \in V_w}$. We say that an F- \mathcal{ALCI} -ontology $T = \{T_i\}_{i \in V}$ is **consistent as witnessed by a module** T_w if T_w^* has a model $\mathcal{I} = \langle \{\mathcal{I}_i\}_{i \in V_w}, \{r_{ij}\}_{(i,j) \in E_w} \rangle$, such that $\Delta^w \neq \emptyset$. A concept C is **satisfiable as witnessed by** T_w if there is a model \mathcal{I} of T_w^* , such that $C^w \neq \emptyset$. A concept subsumption $C \sqsubseteq D$ is **valid as witnessed by** T_w , denoted by $C \sqsubseteq_w D$, if, for every model \mathcal{I} of T_w^* , $C^w \subseteq D^w$. An alternative notation for $C \sqsubseteq_w D$ is $T_w^* \models_w C \sqsubseteq D$.

Example 1. This example illustrates the syntax and semantics of contextualized intersection in F- \mathcal{ALCI} . Suppose that in Module i , there are two concepts, named A and B , correspond-

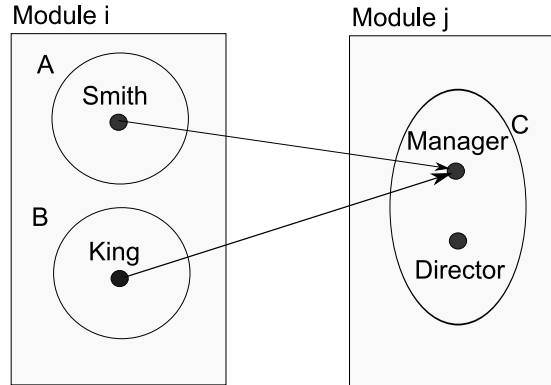


Figure 1.1 Interpretation of Contextualized Intersection.

ing, respectively, to employees of a company with salaries less than \$100,000 and greater than

or equal to \$100,000. (See Figure 1.1.) Module j , on the other hand, has one native concept C corresponding to categories of employees in the company, e.g., administrators, managers, directors, clerks, etc., and it imports concepts A and B from module i . The image domain relation r_{ij} maps both manager **Smith**, who earns a salary of less than \$100,000, and manager **King**, who earns a salary of more than \$100,000, to **Manager**, which is an instance of concept C in j . One may verify that, whereas $(A \sqcap_i B)^j = \emptyset$, we have that $(A \sqcap_j B)^j \neq \emptyset$. This is because the interpretation $(A \sqcap_i B)^j$ asks about employees that are earning, at the same time, less than \$100,000 and at least \$100,000. Clearly, no such employees exist. On the other hand, the interpretation of $(A \sqcap_j B)^j$ tries to classify those categories of employees that contain both individuals of high and individuals of lower salaries. **Manager** is obviously such a category.

Example 2. This second example illustrates the syntax and semantics of contextualized negation in $\mathcal{F}\text{-}\mathcal{ALCT}$. We deal again with two modules i and j and with the same concepts as before. (See Figure 1.2.) In this example we assume that all individuals in the universe are represented in the figure. Concept A contains two employees **Smith** and **Jones**, whereas Concept B contains two employees **King** and **Prince**. Both **Smith** and **King** are managers and both **Jones** and **Prince** are directors. This is reflected in the image domain relation r_{ij} , as illustrated in Figure 1.2. One may verify that, in this case, $(\neg_i A)^j = C^j$, whereas $(\neg_j A)^j = \emptyset$.

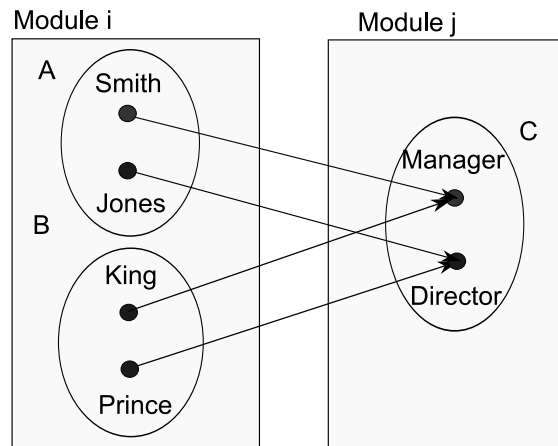


Figure 1.2 Interpretation of Contextualized Negation.

This formal semantical interpretation may be explained by pointing out that from the point

of view of Module j , the interpretation $(\neg_i A)^j$ contains those categories of employees that contain individuals earning at least \$100,000. On the other hand, again from the point of view of Module j , the interpretation of $(\neg_j A)^j$ refers to those categories of employees that do not contain any individual with lower salary. \square

1.3 The Property of Exactness and a Characterization for F- \mathcal{ALCI}

Exactness is a property of *some* interpretations of federated description logics, which ensures seamless propagation of knowledge across importing chains. More precisely, if a concept C in module k is imported by both module i and module j , and module j imports module i , then exactness is equivalent to $r_{kj}(C^k) = r_{ij}(r_{ki}(C^k))$. This has the consequence that, if $\mathcal{I} \models^i C \sqsubseteq D$, then $\mathcal{I} \models^j C \sqsubseteq D$. This is a property that may be very desirable in some contexts but not absolutely necessary in others. Since it imposes rather strong restrictions on the models, we impose it on our interpretations selectively rather than require that it holds universally, as is done in [8].

Definition 4 (Exactness) *Given an edge $(i, j) \in E$, a F- \mathcal{ALCI} -interpretation $\mathcal{I} = \langle \{\mathcal{I}_i\}_{i \in V}, \{r_{ij}\}_{(i,j) \in E} \rangle$ is said to be (i, j) -**exact** if, for every $C \in \widehat{\mathcal{C}}_i \cap \widehat{\mathcal{C}}_j$, $r_{ij}(C^i) = C^j$. \mathcal{I} is **exact** if it is (i, j) -exact, for all $(i, j) \in E$.*

Example: Figure 1.3 depicts an F- \mathcal{ALCI} -interpretation that is not exact. The graph G has three vertices i, j, k and three edges $(k, i), (k, j), (i, j)$. There is one k -concept name A that is imported by both modules i and j and there are no role names. Note that $A^i = r_{ki}(A^k)$ and $A^j = r_{kj}(A^k)$, as required by the definition of interpretation. However, for the concept $\neg_k A \in \widehat{\mathcal{C}}_i \cap \widehat{\mathcal{C}}_j$, we obtain $r_{ij}((\neg_k A)^i) = r_{ij}(r_{ki}(\Delta^k \setminus A^k)) = r_{ij}(r_{ki}(\{y\})) = \emptyset$, whereas $(\neg_k A)^j = r_{kj}(\Delta^k \setminus A^k) = r_{kj}(\{y\}) = \{y''\}$. Thus, the indicated interpretation is not an exact interpretation.

On the other hand, if M_i contained an individual $y' \notin A^i$, such that $(y, y') \in r_{ki}$ and $(y', y'') \in r_{ij}$, then the depicted interpretation would have been exact. \square

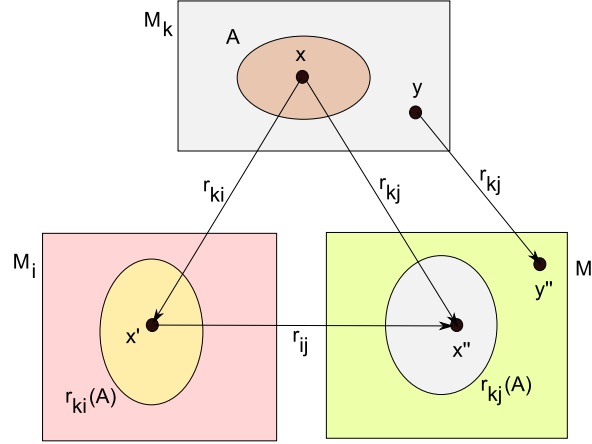


Figure 1.3 A Non-exact Interpretation.

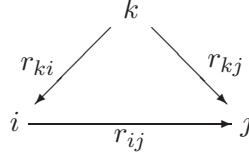
Note that, in general, the notion of exactness in Definition 4 requires that the condition $r_{ij}(C^i) = C^j$ holds for an infinite collection of concept expressions. For our applications the following weaker concept of exactness, that depends on the contents of a specific knowledge base under consideration, suffices. First let us call a set $\mathcal{E}_i \subseteq \widehat{\mathcal{C}}_i$ of i -concept expressions **closed** if it is closed under concept sub-expressions, i.e., for every $C \in \mathcal{E}_i$, all sub-concepts of C are also in \mathcal{E}_i .

Definition 5 (Exactness for T) Let $\mathcal{E} = \{\mathcal{E}_i\}_{i \in V}$, with $\mathcal{E}_i \subseteq \widehat{\mathcal{C}}_i$, $i \in V$, be a V -indexed collection of closed sets of concept expressions and $\mathcal{I} = \langle \{\mathcal{I}_i\}_{i \in V}, \{r_{ij}\}_{(i,j) \in E} \rangle$ be an $F\text{-}\mathcal{ALCI}$ -interpretation. Given $(i,j) \in E$, \mathcal{I} is said to be (i,j) -**exact for \mathcal{E}** if, for every $C \in \mathcal{E}_i \cap \mathcal{E}_j$, $r_{ij}(C^i) = C^j$. \mathcal{I} is **exact for \mathcal{E}** if it is (i,j) -exact for \mathcal{E} , for all $(i,j) \in E$.

Let $T = \{T_i\}_{i \in V}$ be an $F\text{-}\mathcal{ALCI}$ -ontology and $\mathcal{I} = \langle \{\mathcal{I}_i\}_{i \in V}, \{r_{ij}\}_{(i,j) \in E} \rangle$ an $F\text{-}\mathcal{ALCI}$ -interpretation. \mathcal{I} is said to be (i,j) -**exact for T** if it is (i,j) -exact for $\overline{\mathcal{C}} := \{\overline{\mathcal{C}}_i\}_{i \in V}$ and it is said to be **exact for T** if it is exact for $\overline{\mathcal{C}} := \{\overline{\mathcal{C}}_i\}_{i \in V}$.

An alternative condition characterizing the exactness of an $F\text{-}\mathcal{ALCI}$ -interpretation is provided in the following lemma.

Lemma 6 An $F\text{-}\mathcal{ALCI}$ -interpretation $\mathcal{I} = \langle \{\mathcal{I}_i\}_{i \in V}, \{r_{ij}\}_{(i,j) \in E} \rangle$ is exact if and only if, for all $k, i, j \in V$, such that $(k,i), (k,j), (i,j) \in E$, $r_{ij}(r_{ki}(C^k)) = r_{kj}(C^k)$, for every $C \in \widehat{\mathcal{C}}_i \cap \widehat{\mathcal{C}}_j \cap \widehat{\mathcal{C}}_k$. The importing relations are depicted in the following importing diagram.



Proof:

\Rightarrow : If \mathcal{I} is exact, then, for every $C \in \widehat{\mathcal{C}}_i \cap \widehat{\mathcal{C}}_j \cap \widehat{\mathcal{C}}_k$, we have that $r_{ij}(r_{ki}(C^k)) = r_{ij}(C^i) = C^j = r_{kj}(C^k)$.

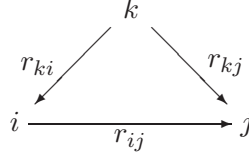
\Leftarrow : For this direction, suppose that, for all $k, i, j \in V$, such that $(k, i), (k, j), (i, j) \in E$, $r_{ij}(r_{ki}(B^k)) = r_{kj}(B^k)$, for every i -, j - and k -concept B . Consider $C \in \widehat{\mathcal{C}}_i \cap \widehat{\mathcal{C}}_j$. To show that $r_{ij}(C^i) = C^j$, we apply structural induction on C . We have:

- $r_{ij}(\top_k^i) = r_{ij}(r_{ki}(\Delta^k)) = r_{kj}(\Delta^k) = \top_k^j$.
- $r_{ij}(C^i) = r_{ij}(r_{ki}(C^k)) = r_{kj}(C^k) = C^j$, for every $C \in \mathcal{C}_k \cap \widehat{\mathcal{C}}_i \cap \widehat{\mathcal{C}}_j$.
- $r_{ij}((\neg_k C)^i) = r_{ij}(r_{ki}(\Delta^k - C^k)) = r_{ij}(r_{ki}((\neg_k C)^k)) = r_{kj}((\neg_k C)^k) = r_{kj}(\Delta^k - C^k) = (\neg_k C)^j$.
- $r_{ij}((C \sqcap_k D)^i) = r_{ij}(r_{ki}(C^k \cap D^k)) = r_{ij}(r_{ki}((C \sqcap_k D)^k)) = r_{kj}((C \sqcap_k D)^k) = r_{kj}(C^k \cap D^k) = (C \sqcap_k D)^j$.
- $r_{ij}((\exists_k R.C)^i) = r_{ij}(r_{ki}((\exists_k R.C)^k)) = r_{kj}((\exists_k R.C)^k) = (\exists_k R.C)^j$.

The cases of $C \sqcup_k D$ and of $\forall_k R.C$ are handled similarly. ■

Employing the same proof, but with “exact for \mathcal{E} ” in place of “exact”, we obtain the following lemma providing a necessary and sufficient condition for the exactness of an F- \mathcal{ALCI} -interpretation for a given V -indexed collection \mathcal{E} of closed sets of concept expressions.

Lemma 7 *Let $\mathcal{E} = \{\mathcal{E}_i\}_{i \in V}$, with $\mathcal{E}_i \subseteq \widehat{\mathcal{C}}_i$, $i \in V$, be a V -indexed collection of closed sets of concept expressions and $\mathcal{I} = \langle \{\mathcal{I}_i\}_{i \in V}, \{r_{ij}\}_{(i,j) \in E} \rangle$ an F- \mathcal{ALCI} -interpretation. \mathcal{I} is exact for \mathcal{E} if and only if, for all $k, i, j \in V$, such that $(k, i), (k, j), (i, j) \in E$, $r_{ij}(r_{ki}(C^k)) = r_{kj}(C^k)$, for every $C \in \mathcal{E}_i \cap \mathcal{E}_j \cap \mathcal{E}_k$. The importing relations are depicted in the following importing diagram.*



Based on the definition of an exact interpretation, we define exact models of an F- \mathcal{ALCI} -ontology.

Definition 8 (Exact Model) Let $T = \{T_i\}_{i \in V}$ be an F- \mathcal{ALCI} -ontology. An interpretation $\mathcal{I} = \langle \{\mathcal{I}_i\}_{i \in V}, \{r_{ij}\}_{(i,j) \in E} \rangle$ is an **exact model** of T if it is exact for T and $\mathcal{I} \models T$. T is said to be **exactly consistent as witnessed by a module** T_w if there exists an exact model \mathcal{I} of T_w^* , such that $\Delta^w \neq \emptyset$. A concept C is **exactly satisfiable as witnessed by** T_w if there exists an exact model \mathcal{I} of T_w^* , such that $C^w \neq \emptyset$. Finally, a concept subsumption $C \sqsubseteq D$ is **exactly valid as witnessed by** T_w , denoted $C \sqsubseteq_w^e D$ if, for every exact model \mathcal{I} of T_w^* , $C^w \subseteq D^w$. In this case we also write $T_w^* \models_w^e C \sqsubseteq D$.

1.4 A Reduction from F- \mathcal{ALCI} to \mathcal{ALCI}

We establish the decidability of F- \mathcal{ALCI} , by providing a reduction \mathfrak{R} from an F- \mathcal{ALCI} KB $\Sigma_d = \{T_i\}$ to an \mathcal{ALCI} KB $\Sigma := \mathfrak{R}(\Sigma_d)$. Whereas the reduction from a distributed model in the case of P-DL [8] was accomplished by creating equivalence classes of individuals related via image domain relations, our reduction is based on taking a disjoint union of local domains and simulating the image domain relations via newly introduced binary relations.

The **reduction** \mathfrak{R} from an F- \mathcal{ALCI} KB $\Sigma_d = \{T_i\}$ to an \mathcal{ALCI} KB $\Sigma := \mathfrak{R}(\Sigma_d)$ is obtained as follows:

The signature of Σ is the union of the local signatures of the modules together with a global top \top , a global bottom \perp , local top concepts \top_i , for all $i \in V$, and, finally, a collection of new role names $\{R_{ij}\}_{(i,j) \in E}$, i.e.,

$$\text{Sig}(\Sigma) = \bigcup_i (\mathcal{C}_i \cup \mathcal{R}_i) \cup \{\top, \perp\} \cup \{\top_i : 1 \leq i \leq n\} \cup \{R_{ij} : (i, j) \in E\}.$$

Moreover, various axioms derived from the structure of Σ_d are added to Σ .

- For each $C \in \mathcal{C}_i$, $C \sqsubseteq \top_i$ is added to Σ .
- For each $R \in \mathcal{R}_i$, \top_i is stipulated to be the domain and range of R , i.e., $\top \sqsubseteq \forall R^-. \top_i$ and $\top \sqsubseteq \forall R. \top_i$ are added to Σ .
- For each new role name R_{ij} , \top_i is stipulated to be its domain and \top_j to be its range, i.e., $\top \sqsubseteq \forall R_{ij}^-. \top_i$ and $\top \sqsubseteq \forall R_{ij}. \top_j$ are added to Σ .
- For each $C \sqsubseteq D \in T_i$, $\#_i(C) \sqsubseteq \#_i(D)$ is added to Σ , where $\#_i$ is a function from $\widehat{\mathcal{C}}_i$ to the set of \mathcal{ALCT} -concepts; It serves to maintain the compatibility of the concept domains inside the “unified” domain. The precise definition of $\#_i$ is given below.

If, in addition to the previous conditions, for every importing diagram of the form

$$\begin{array}{ccc}
 & k & \\
 i & \swarrow \quad \searrow & j \\
 i & \xrightarrow{\quad} & j
 \end{array} \tag{1.2}$$

and all $C \in \overline{\mathcal{C}}_i \cap \overline{\mathcal{C}}_j \cap \overline{\mathcal{C}}_k$, $\exists R_{ij}^-. (\exists R_{ki}^-. \#_k(C)) = \exists R_{kj}^-. \#_k(C)$ is added to Σ , then the reduction is said to be an **exact reduction** and is denoted by $\mathfrak{R}_e(\Sigma_d)$.

The mapping $\#_i(C)$ is defined by induction on the structure of $C \in \widehat{\mathcal{C}}_i$, where we use the notation $\mathcal{R}_k^- := \{R^- : R \in \mathcal{R}_k\}$:

- $\#_i(C) = C$, if $C \in \mathcal{C}_i$;
- $\#_i(C) = \exists R_{ji}^-. \#_j(C)$, if $C \in \mathcal{C}_j \cap \widehat{\mathcal{C}}_i$;
- $\#_i(\neg_j D) = \exists R_{ji}^-. (\neg \#_j(D) \sqcap \top_j)$;
- $\#_i(D \boxplus_j E) = \exists R_{ji}^-. (\#_j(D) \boxplus \#_j(E))$, where $\boxplus = \sqcap$ or $\boxplus = \sqcup$;
- $\#_i(\exists_j R. D) = \exists R_{ji}^-. (\exists R_{kj}^-. (\exists R. (\exists R_{kj}. \#_j(D))))$, if $R \in \mathcal{R}_k \cup \mathcal{R}_k^-$;
- $\#_i(\forall_j R. D) = \exists R_{ji}^-. (\forall R_{kj}^-. (\forall R. (\forall R_{kj}. \#_j(D))))$, if $R \in \mathcal{R}_k \cup \mathcal{R}_k^-$.

The translation $\#_i$ ensures that the locality of concept expressions and the interaction between local domains in the distributed model are adequately reflected in the unified domain.

It will be shown that the reduction \mathfrak{R} is sound and complete in the sense that, if the local top concept \top_w in $\mathfrak{R}(\Sigma_d)$, that corresponds to a module T_w in Σ_d , is satisfiable in an \mathcal{ALCI} -model of $\mathfrak{R}(\Sigma_d)$, then Σ_d itself is consistent as witnessed by T_w and vice-versa. Soundness will be taken up in the next section and completeness in Section 6.

1.5 Soundness of the Reduction

The goal of this section is to prove that the translation \mathfrak{R} is sound. Roughly speaking, it is shown that, if the translation of a distributed KB under \mathfrak{R} has a model, then the distributed KB itself has a model.

Let Σ_d be an F- \mathcal{ALCI} KB, and T_w a module of Σ_d . If \top_w is satisfiable with respect to $\mathfrak{R}(T_w^*)$, then Σ_d is consistent as witnessed by T_w .

Definition 9 Let $\Sigma_d = \{T_i\}_{i \in V}$ be an F- \mathcal{ALCI} KB and $\mathcal{I} = \langle \Delta^{\mathcal{I}}, \cdot^{\mathcal{I}} \rangle$ an interpretation of the \mathcal{ALCI} KB $\mathfrak{R}(\Sigma_d)$. Construct an interpretation $\mathbf{F}(\mathcal{I}) = \langle \{\mathcal{I}_i\}_{i \in V}, \{r_{ij}\}_{(i,j) \in E} \rangle$ for Σ_d as follows:

- $\Delta^i = \top_i^{\mathcal{I}}$, for all $i \in V$;
- $C^i = C^{\mathcal{I}}$, for every $C \in \mathcal{C}_i$;
- $R^i = R^{\mathcal{I}}$, for every $R \in \mathcal{R}_i$;
- $r_{ij} = R_{ij}^{\mathcal{I}}$, for every $(i, j) \in E$.

We start with a technical lemma that shows, roughly speaking, that the image of the interpretation of a concept C under the interpretation of one of the new role names R_{ij} is equal to the interpretation of the concept $\exists R_{ij}^-.C$, all in the same model. This lemma is preparatory in dealing with the various cases involved in the definition of the translation function $\#_i$.

Lemma 10 Let Σ_d be an F- \mathcal{ALCI} KB and $\mathcal{I} = \langle \Delta^{\mathcal{I}}, \cdot^{\mathcal{I}} \rangle$ an interpretation for $\mathfrak{R}(\Sigma_d)$. Then, for every concept $C \in \widehat{\mathcal{C}}_i$, such that $\exists R_{ij}^-.C$ occurs in $\mathfrak{R}(\Sigma_d)$,

$$R_{ij}^{\mathcal{I}}(C^{\mathcal{I}}) = (\exists R_{ij}^-.C)^{\mathcal{I}}.$$

Proof:

We have

$$\begin{aligned}
(\exists R_{ij}^-.C)^{\mathcal{I}} &= \{x \in \Delta^{\mathcal{I}} : (\exists y \in C^{\mathcal{I}})((x, y) \in R_{ij}^{\mathcal{I}-})\} \quad (\text{by the defin. of } \cdot^{\mathcal{I}}) \\
&= \{x \in \Delta^{\mathcal{I}} : (\exists y \in C^{\mathcal{I}})((y, x) \in R_{ij}^{\mathcal{I}})\} \quad (\text{by the defin. of } R_{ij}^{\mathcal{I}-}) \\
&= R_{ij}^{\mathcal{I}}(C^{\mathcal{I}}). \quad (\text{by the definition of } R_{ij}^{\mathcal{I}}(C^{\mathcal{I}}))
\end{aligned}$$

■

Next, we present another technical lemma to the effect that the interpretation of the concept $\forall R_{kj}^-. (\forall R. (\forall R_{kj}. \#_j(C)))$ formed using the translation $\#_j(C)$ of a concept $C \in \widehat{\mathcal{C}}_j$ and the role name $R \in \mathcal{R}_k$, equals to

$$\forall R_{kj}^{\mathcal{I}}(R^{\mathcal{I}}). \#_j(C)^{\mathcal{I}} := \{x \in \top_j^{\mathcal{I}} : (\forall y \in \top_j^{\mathcal{I}})((x, y) \in R_{kj}^{\mathcal{I}}(R^{\mathcal{I}}) \rightarrow y \in \#_j(C)^{\mathcal{I}})\}.$$

This lemma will help us deal with the universal quantification case involved in the recursive definition of the translation function $\#_i$.

Lemma 11 *Let Σ_d be an $F\text{-}\mathcal{ALCI}$ KB and $\mathcal{I} = \langle \Delta^{\mathcal{I}}, \cdot^{\mathcal{I}} \rangle$ an interpretation for $\mathfrak{R}(\Sigma_d)$. Then, for all $C \in \widehat{\mathcal{C}}_j$, $R \in \mathcal{R}_k$, such that $\forall R_{kj}^-. (\forall R. (\forall R_{kj}. \#_j(C)))$ occurs in $\mathfrak{R}(\Sigma_d)$*

$$(\forall R_{kj}^-. (\forall R. (\forall R_{kj}. \#_j(C))))^{\mathcal{I}} = \forall R_{kj}^{\mathcal{I}}(R^{\mathcal{I}}). \#_j(C)^{\mathcal{I}}.$$

Proof:

For the left-to-right inclusion, suppose that $x \in (\forall R_{kj}^-. (\forall R. (\forall R_{kj}. \#_j(C))))^{\mathcal{I}}$. The following diagrams help illustrate the argument.

$$\begin{array}{ccc}
x & \xleftarrow{R_{kj}^{\mathcal{I}}} & y \\
& & \downarrow R^{\mathcal{I}} \\
w & \xleftarrow{R_{kj}^{\mathcal{I}}} & z
\end{array}
\qquad
\begin{array}{ccc}
x & \xleftarrow{R_{kj}^{\mathcal{I}}} & u \\
& & \downarrow R^{\mathcal{I}} \\
v & \xleftarrow{R_{kj}^{\mathcal{I}}} & t
\end{array}$$

Then, for all $y \in \Delta^{\mathcal{I}}$, with $(y, x) \in R_{kj}^{\mathcal{I}}$, we must have

$$y \in (\forall R.(\forall R_{kj}.\#_j(C)))^{\mathcal{I}}.$$

Thus, for all $z \in \Delta^{\mathcal{I}}$, such that $(y, z) \in R^{\mathcal{I}}$, we have that $z \in (\forall R_{kj}.\#_j(C))^{\mathcal{I}}$, i.e., for all $w \in \Delta^{\mathcal{I}}$, such that $(z, w) \in R_{kj}^{\mathcal{I}}$, $w \in \#_j(C)^{\mathcal{I}}$, see the left diagram above.

Now assume that $(x, v) \in R_{kj}^{\mathcal{I}}(R^{\mathcal{I}})$, for some $v \in \Delta^{\mathcal{I}}$. Then, there exist $u, t \in \Delta^{\mathcal{I}}$, such that $(u, t) \in R^{\mathcal{I}}$, $(u, x), (t, v) \in R_{kj}^{\mathcal{I}}$, see the right diagram above. Then, by what was shown in the previous paragraph, $v \in \#_j(C)^{\mathcal{I}}$, whence we conclude that $x \in \forall R_{kj}^{\mathcal{I}}(R^{\mathcal{I}}).\#_j(C)^{\mathcal{I}}$.

For the right-to-left inclusion, assume that $x \in \forall R_{kj}^{\mathcal{I}}(R^{\mathcal{I}}).\#_j(C)^{\mathcal{I}}$. Thus, for all $v \in \Delta^{\mathcal{I}}$, such that $(x, v) \in R_{kj}^{\mathcal{I}}(R^{\mathcal{I}})$, we must have $v \in \#_j(C)^{\mathcal{I}}$. Now assume that $(y, x) \in R_{kj}^{\mathcal{I}}$, $(y, z) \in R^{\mathcal{I}}$ and $(z, w) \in R_{kj}^{\mathcal{I}}$. This implies that $(x, w) \in R_{kj}^{\mathcal{I}}(R^{\mathcal{I}})$. Thus, $w \in \#_j(C)^{\mathcal{I}}$. This proves that $x \in (\forall R_{kj}^{\mathcal{I}}.\forall R.(\forall R_{kj}.\#_j(C)))^{\mathcal{I}}$. ■

To connect the interpretation \mathcal{I} with its federated counterpart $\mathbf{F}(\mathcal{I})$, we need to establish a correspondence between the interpretation of the translation $\#_i(C)$ of a concept $C \in \widehat{\mathcal{C}}_i$ under \mathcal{I} and that of the concept C under $\mathbf{F}(\mathcal{I})$. This relationship is explored in the following lemma.

Lemma 12 *Let Σ_d be an $F\text{-}\mathcal{ALCI}$ KB, $\mathcal{I} = \langle \Delta^{\mathcal{I}}, \cdot^{\mathcal{I}} \rangle$ an interpretation for $\mathfrak{R}(\Sigma_d)$ and $\mathbf{F}(\mathcal{I}) = \langle \{\mathcal{I}_i\}_{i \in V}, \{r_{ij}\}_{(i,j) \in E} \rangle$, with $\mathcal{I}_i = \langle \Delta^i, \cdot^i \rangle, i \in V$. Then*

$$\#_i(C)^{\mathcal{I}} = C^i, \quad \text{for every } C \in \widehat{\mathcal{C}}_i, i \in V.$$

Proof:

The proof is by structural induction on C .

For the basis of the induction, if $C \in \mathcal{C}_i$,

$$\begin{aligned} \#_i(C)^{\mathcal{I}} &= C^{\mathcal{I}} \quad (\text{by the definition of } \#_i(C)) \\ &= C^i, \quad (\text{by the definition of } C^i) \end{aligned}$$

whereas, if $C \in \mathcal{C}_j \cap \widehat{\mathcal{C}}_i$,

$$\begin{aligned}
\#_i(C)^{\mathcal{I}} &= (\exists R_{ji}^-. \#_j(C))^{\mathcal{I}} \quad (\text{by the definition of } \#_i(C)) \\
&= R_{ji}^{\mathcal{I}}(\#_j(C)^{\mathcal{I}}) \quad (\text{by Lemma 10}) \\
&= r_{ji}(C^j) \quad (\text{by the definition of } r_{ji} \text{ and the previous case}) \\
&= C^i. \quad (\text{by the definition of } C^i)
\end{aligned}$$

For $C = \neg_j D$, we have

$$\begin{aligned}
\#_i(\neg_j D)^{\mathcal{I}} &= (\exists R_{ji}^-. (\neg \#_j(D) \sqcap \top_j))^{\mathcal{I}} \quad (\text{by the definition of } \#_i(\neg_j D)) \\
&= R_{ji}^{\mathcal{I}}((\neg \#_j(D) \sqcap \top_j)^{\mathcal{I}}) \quad (\text{by Lemma 10}) \\
&= R_{ji}^{\mathcal{I}}((\neg \#_j(D))^{\mathcal{I}} \sqcap \top_j^{\mathcal{I}}) \quad (\text{by the definition of } \cdot^{\mathcal{I}}) \\
&= R_{ji}^{\mathcal{I}}((\Delta^{\mathcal{I}} \setminus \#_j(D)^{\mathcal{I}}) \sqcap \top_j^{\mathcal{I}}) \quad (\text{by the definition of } \cdot^{\mathcal{I}}) \\
&= r_{ji}((\Delta^{\mathcal{I}} \setminus D^j) \sqcap \Delta^j) \quad (\text{by the definition of } F(\mathcal{I}) \\
&\quad \text{and the induction hypothesis}) \\
&= r_{ji}(\Delta^j \setminus D^j) \quad (\text{set-theoretically}) \\
&= (\neg_j D)^i. \quad (\text{by the definition of } (\neg_j D)^i)
\end{aligned}$$

For $\boxplus = \sqcap$ or $\boxplus = \sqcup$, and denoting by $\oplus = \cap$ or $\oplus = \cup$, respectively, the corresponding set-theoretic operation,

$$\begin{aligned}
\#_i(D \boxplus_j E)^{\mathcal{I}} &= (\exists R_{ji}^-. (\#_j(D) \boxplus \#_j(E)))^{\mathcal{I}} \quad (\text{by the definition of } \#_i(D \boxplus_j E)^{\mathcal{I}}) \\
&= R_{ji}^{\mathcal{I}}((\#_j(D) \boxplus \#_j(E))^{\mathcal{I}}) \quad (\text{by Lemma 10}) \\
&= R_{ji}^{\mathcal{I}}(\#_j(D)^{\mathcal{I}} \oplus \#_j(E)^{\mathcal{I}}) \quad (\text{by the definition of } \cdot^{\mathcal{I}}) \\
&= r_{ji}(D^j \oplus E^j) \quad (\text{by the definition of } F(\mathcal{I}) \\
&\quad \text{and the induction hypothesis}) \\
&= (D \boxplus_j E)^i. \quad (\text{by the definition of } (D \boxplus_j E)^i)
\end{aligned}$$

For $C = \#_i(\exists_j R.D)$, with $R \in \mathcal{R}_k$, we first show that

$$r_{kj}(R^{\mathcal{I}-}(r_{kj}^-(D^j))) = \{x \in \Delta^j : (\exists w \in D^j)((x, w) \in R^j)\}. \quad (1.3)$$

For the left-to-right inclusion, assume that $x \in r_{kj}(R^{\mathcal{I}-}(r_{kj}^-(D^j)))$. Then, there exists $y \in R^{\mathcal{I}-}(r_{kj}^-(D^j))$, such that $(y, x) \in r_{kj}$ and a $z \in r_{kj}^-(D^j)$, such that $(y, z) \in R^{\mathcal{I}}$. Hence, there exists $w \in D^j$, such that $(z, w) \in r_{kj}$. These relations are depicted in the following diagram.

$$\begin{array}{ccc} y & \xrightarrow{r_{kj}} & x \\ R^{\mathcal{I}} \downarrow & & \\ z & \xrightarrow{r_{kj}} & w \end{array}$$

This shows that $(x, w) \in r_{kj}(R^{\mathcal{I}}) = r_{kj}(R^k) = R^j$ and, as a result, that $x \in \{t \in \Delta^j : (\exists w \in D^j)((t, w) \in R^j)\}$.

Suppose, for the reverse inclusion, that $x \in \{t \in \Delta^j : (\exists w \in D^j)((t, w) \in R^j)\}$. Thus, there exists $w \in D^j$, such that $(x, w) \in R^j = r_{kj}(R^k) = r_{kj}(R^{\mathcal{I}})$. Hence, for some $(y, z) \in R^{\mathcal{I}}$, we have $(y, x) \in r_{kj}$ and $(w, z) \in r_{kj}^-$. This shows that $x \in r_{kj}(y) \subseteq r_{kj}(R^{\mathcal{I}-}(z)) \subseteq r_{kj}(R^{\mathcal{I}-}(r_{kj}^-(w))) \subseteq r_{kj}(R^{\mathcal{I}-}(r_{kj}^-(D^j)))$ and concludes the proof of Equation (1.3).

Now we get that

$$\begin{aligned} \#_i(\exists_j R.D)^{\mathcal{I}} &= (\exists R_{ji}^-. (\exists R_{kj}^-. (\exists R. (\exists R_{kj}. (\#_j(D))))))^{\mathcal{I}} \\ &\quad (\text{by the definition of } \#_i(\exists_j R.D)) \\ &= R_{ji}^{\mathcal{I}}(R_{kj}^{\mathcal{I}}(R^{\mathcal{I}-}(R_{kj}^{\mathcal{I}-}(\#_j(D)^{\mathcal{I}})))) \quad (\text{by Lemma 10}) \\ &= r_{ji}(r_{kj}(R^{\mathcal{I}-}(r_{kj}^-(D^j)))) \quad (\text{by the definition of } \mathbf{F}(\mathcal{I})) \\ &\quad \text{and the induction hypothesis} \\ &= r_{ji}(\{x \in \Delta^j : (\exists y \in D^j)((x, y) \in R^j)\}) \quad (\text{by Equation (1.3)}) \\ &= (\exists_j R.D)^i. \quad (\text{by the definition of } (\exists_j R.D)^i) \end{aligned}$$

For $C = \#_i(\forall_j R.D)$, with $R \in \mathcal{R}_k$, recall that, by Lemma 11,

$$\begin{aligned} (\forall R_{kj}^-. (\forall R. (\forall R_{kj}. (\#_j(D))))))^{\mathcal{I}} = \\ \{x \in \top_j^{\mathcal{I}} : (\forall y \in \top_j^{\mathcal{I}})((x, y) \in R_{kj}^{\mathcal{I}}(R^{\mathcal{I}}) \rightarrow y \in \#_j(D)^{\mathcal{I}})\}. \end{aligned} \quad (1.4)$$

Therefore, we have

$$\begin{aligned} \#_i(\forall_j R.D)^{\mathcal{I}} &= (\exists R_{ji}^-. (\forall R_{kj}^-. (\forall R. (\forall R_{kj}. (\#_j(D))))))^{\mathcal{I}} \\ &\quad \text{(by the definition of } \#_i(\forall_j R.D) \text{)} \\ &= R_{ji}^{\mathcal{I}}((\forall R_{kj}^-. (\forall R. (\forall R_{kj}. (\#_j(D))))))^{\mathcal{I}} \quad \text{(by Lemma 10)} \\ &= R_{ji}^{\mathcal{I}}(\{x \in \top_j^{\mathcal{I}} : (\forall y \in \top_j^{\mathcal{I}})((x, y) \in R_{kj}^{\mathcal{I}}(R^{\mathcal{I}}) \rightarrow y \in \#_j(D)^{\mathcal{I}})\}) \\ &\quad \text{(by Equation (1.4))} \\ &= r_{ji}(\{x \in \Delta^j : (\forall y \in \Delta^j)((x, y) \in R^j \rightarrow y \in D^j)\}) \\ &\quad \text{(by the definition of } \mathbf{F}(\mathcal{I}) \text{ and the induction hypothesis)} \\ &= (\forall_j R.D)^i. \quad \text{(by the definition of } (\forall_j R.D)^i \text{)} \end{aligned}$$

■

The following is the soundness theorem for the reduction \mathfrak{R} .

Theorem 13 (Soundness) *Let Σ_d be an $F\text{-}\mathcal{ALCC}$ KB, and T_w a module of Σ_d . If \top_w is satisfiable with respect to $\mathfrak{R}(T_w^*)$, then Σ_d is consistent as witnessed by T_w .*

Proof:

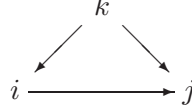
Suppose that \top_w is satisfiable with respect to $\mathfrak{R}(T_w^*)$. Then $\mathfrak{R}(T_w^*)$ has a model $\mathcal{I} = \langle \Delta^{\mathcal{I}}, \cdot^{\mathcal{I}} \rangle$, such that $\top_w^{\mathcal{I}} \neq \emptyset$. Our goal is to show that $\mathbf{F}(\mathcal{I}) = \langle \{\mathcal{I}_i\}_{i \in V_w}, \{r_{ij}\}_{(i,j) \in E_w} \rangle$ is a model of T_w^* , such that $\Delta^w \neq \emptyset$.

Clearly, we have $\Delta^w = \top_w^{\mathcal{I}} \neq \emptyset$, by the hypothesis. So it suffices to show that $\mathbf{F}(\mathcal{I})$ is a model of the federated ontology T_w^* , i.e., that it satisfies $\mathcal{I}_i \models T_i$, for every $i \in V_w$. Suppose that $C \sqsubseteq D \in T_i$. By the construction of $\mathfrak{R}(T_w^*)$ and the fact that $\mathcal{I} \models \mathfrak{R}(T_w^*)$, we must have $\#_i(C)^{\mathcal{I}} \subseteq \#_i(D)^{\mathcal{I}}$, whence, by Lemma 12, we obtain that $C^i \subseteq D^i$, showing that $\mathbf{F}(\mathcal{I}) \models T_w^*$.

■

To establish the soundness in the case of an *exact* reduction we need to ensure that the federated model $F(\mathcal{I})$ obtained from the model \mathcal{I} of $\mathfrak{R}_e(\Sigma_d)$ is an exact model of Σ_d . Preliminary work towards this goal is accomplished in the following lemma.

Lemma 14 *Let Σ_d be an F-ALCII ontology and $\mathcal{I} = \langle \Delta^{\mathcal{I}}, \cdot^{\mathcal{I}} \rangle$ be a model of $\mathfrak{R}_e(\Sigma_d)$. Then, for every importing diagram*



and all $C \in \overline{\mathcal{C}}_i \cap \overline{\mathcal{C}}_j \cap \overline{\mathcal{C}}_k$, $r_{ij}(r_{ki}(C^k)) = r_{kj}(C^k)$ holds in $F(\mathcal{I})$.

Proof:

$$\begin{aligned}
 r_{ij}(r_{ki}(C^k)) &= R_{ij}^{\mathcal{I}}(R_{ki}^{\mathcal{I}}(\#_k(C)^{\mathcal{I}})) \quad (\text{by the definition of } F(\mathcal{I}) \\
 &\quad \text{and Lemma 12}) \\
 &= (\exists R_{ij}^-. (\exists R_{ki}^-. (\#_k(C))))^{\mathcal{I}} \quad (\text{by Lemma (10)}) \\
 &= (\exists R_{kj}^-. (\#_k(C)))^{\mathcal{I}} \quad (\text{because } \mathcal{I} \models \mathfrak{R}_e(\Sigma_d)) \\
 &= R_{kj}^{\mathcal{I}}(\#_k(C)^{\mathcal{I}}) \quad (\text{by Lemma (10)}) \\
 &= r_{kj}(C^k). \quad (\text{by the definition of } F(\mathcal{I}) \text{ and Lemma 12})
 \end{aligned}$$

■

Finally, we formulate and present the main result on the exact soundness of the translation \mathfrak{R}_e .

Theorem 15 (Exact Soundness) *Let Σ_d be an F-ALCII KB, and T_w a module of Σ_d . If \top_w is satisfiable with respect to $\mathfrak{R}_e(T_w^*)$, then Σ_d is exactly consistent as witnessed by T_w .*

Proof:

Suppose that \top_w is satisfiable with respect to $\mathfrak{R}_e(T_w^*)$. Then $\mathfrak{R}_e(T_w^*)$ has a model $\mathcal{I} = \langle \Delta^{\mathcal{I}}, \cdot^{\mathcal{I}} \rangle$, such that $\top_w^{\mathcal{I}} \neq \emptyset$. Our goal is to show that $F(\mathcal{I})$ is an exact model of T_w^* , such that $\Delta^w \neq \emptyset$.

As before, we have $\Delta^w = \top_w^{\mathcal{I}} \neq \emptyset$, by the hypothesis. So it suffices to show that $F(\mathcal{I})$ is an exact model of T_w^* , i.e., that it satisfies the two conditions postulated in Definition

8. This amounts to showing that (i) for all $k, i, j \in V_w$, such that $(k, i), (k, j), (i, j) \in E_w$, $r_{ij}(r_{ki}(C^k)) = r_{kj}(C^k)$, for every $C \in \overline{\mathcal{C}}_i \cap \overline{\mathcal{C}}_j \cap \overline{\mathcal{C}}_k$, and (ii) $\mathcal{I} \models T_i$, for every $i \in V_w$. The first condition holds by Lemma 14. Condition (ii) may be shown exactly as in the proof of Theorem 13. It follows that Σ_d is exactly consistent as witnessed by T_w . \blacksquare

1.6 Completeness of the Reduction

We turn now to the proof of the completeness of the reduction \mathfrak{R} . Informally speaking, it will be shown that, if an F- \mathcal{ALCI} KB $\Sigma_d = \{T_i\}_{i \in V}$ is consistent as witnessed by a module T_w , then the corresponding local top concept \top_w in $\Sigma = \mathfrak{R}(\Sigma_d)$ is satisfiable. Moreover, we will obtain a correspondence between exact consistency of Σ_d as witnessed by T_w and satisfiability of \top_w in $\mathfrak{R}_e(\Sigma_d)$.

Definition 16 *Suppose that Σ_d is an F- \mathcal{ALCI} KB and that $\mathcal{I}_d = \langle \{\mathcal{I}_i\}_{i \in V}, \{r_{ij}\}_{(i,j) \in E} \rangle$, with $\mathcal{I}_i = \langle \Delta^i, \cdot^i \rangle$, is a model of Σ_d . Construct an interpretation $\mathcal{I} := \mathbf{G}(\mathcal{I}_d) = \langle \Delta^{\mathcal{I}}, \cdot^{\mathcal{I}} \rangle$ of $\mathfrak{R}(\Sigma_d)$ as follows:*

- $\Delta^{\mathcal{I}} = \bigcup_{i \in V} \Delta^i$;
- $\top_i^{\mathcal{I}} = \Delta^i$, for every $i \in V$;
- $C^{\mathcal{I}} = C^i$, for every $C \in \mathcal{C}_i$;
- $R^{\mathcal{I}} = R^i$, for every $R \in \mathcal{R}_i$;
- $R_{ij}^{\mathcal{I}} = r_{ij}$, for every $(i, j) \in E$.

To connect the federated interpretation \mathcal{I}_d with its single module-counterpart $\mathcal{I} := \mathbf{G}(\mathcal{I}_d)$, we need to establish a correspondence between the interpretation of the translation $\#_i(C)$ of a concept $C \in \widehat{\mathcal{C}}_i$ under $\mathbf{G}(\mathcal{I}_d)$ and that of the concept C under \mathcal{I} . Such a correspondence is formulated in the following lemma. In the proof, instead of working by induction on the structure of a concept formula, we show that $\mathcal{I}_d = \mathbf{F}(\mathbf{G}(\mathcal{I}_d))$ and, then, apply Lemma 12.

Lemma 17 *Let Σ_d be an $F\text{-}\mathcal{ALCI}$ KB, $\mathcal{I}_d = \langle \{\mathcal{I}_i\}_{i \in V}, \{r_{ij}\}_{(i,j) \in E} \rangle$ a model of Σ_d and set $\mathcal{I} := \mathbf{G}(\mathcal{I}_d) = \langle \Delta^{\mathcal{I}}, \cdot^{\mathcal{I}} \rangle$. Then*

$$\#_i(C)^{\mathcal{I}} = C^i, \quad \text{for every } C \in \widehat{\mathcal{C}}_i, \ i \in V.$$

Proof:

This will follow directly from Lemma 12 once it is shown that $\mathcal{I}_d = \mathbf{F}(\mathbf{G}(\mathcal{I}_d))$. We have, using the full model names to keep notation clear,

- For all $i \in V$, $\Delta^{\mathbf{F}(\mathbf{G}(\mathcal{I}_d))i} = \top_i^{\mathbf{G}(\mathcal{I}_d)} = \Delta^{(\mathcal{I}_d)i}$.
- For every $C \in \mathcal{C}_i$, $C^{\mathbf{F}(\mathbf{G}(\mathcal{I}_d))i} = C^{\mathbf{G}(\mathcal{I}_d)} = C^{(\mathcal{I}_d)i}$.
- For every $R \in \mathcal{R}_i$, $R^{\mathbf{F}(\mathbf{G}(\mathcal{I}_d))i} = R^{\mathbf{G}(\mathcal{I}_d)} = R^{(\mathcal{I}_d)i}$.
- For every $(i, j) \in E$, $r_{ij}^{[\mathbf{F}(\mathbf{G}(\mathcal{I}_d))]} = R_{ij}^{\mathbf{G}(\mathcal{I}_d)} = r_{ij}^{[\mathcal{I}_d]} = r_{ij}$, where the bracketed superscripts of r_{ij} 's specify which model they are part of.

Therefore, we do indeed have $\mathcal{I}_d = \mathbf{F}(\mathbf{G}(\mathcal{I}_d))$. Hence, by Lemma 12,

$$\#_i(C)^{\mathcal{I}} = \#_i(C)^{\mathbf{G}(\mathcal{I}_d)} = C^{\mathbf{F}(\mathbf{G}(\mathcal{I}_d))i} = C^{(\mathcal{I}_d)i} = C^i.$$

■

The main goal of this section is to show that the converse of Theorem 13 also holds. More precisely, we have the following

Theorem 18 (Completeness) *Let $\Sigma_d = \{T_i\}_{i \in V}$ be an $F\text{-}\mathcal{ALCI}$ ontology. If Σ_d is consistent as witnessed by a module T_w , then \top_w is satisfiable with respect to $\mathfrak{R}(T_w^*)$.*

Proof:

Suppose that Σ_d is consistent as witnessed by T_w . Thus, it has a model $\mathcal{I}_d = \langle \{\mathcal{I}_i\}_{i \in V}, \{r_{ij}\}_{(i,j) \in E} \rangle$, such that $\Delta^w \neq \emptyset$. We proceed to show that $\mathcal{I} := \mathbf{G}(\mathcal{I}_d)$ is a model of $\mathfrak{R}(T_w^*)$, such that $\top_w^{\mathcal{I}} \neq \emptyset$. We have $\top_w^{\mathcal{I}} = \Delta^w \neq \emptyset$, by the hypothesis. We show, next, that all axioms in $\mathfrak{R}(T_w^*)$ are satisfied by $\mathbf{G}(\mathcal{I}_d)$.

Clearly, if $C \in \mathcal{C}_i$, then $C^{\mathcal{I}} = C^i \subseteq \Delta^i = \top_i^{\mathcal{I}}$, whence $C \sqsubseteq \top_i$ holds in \mathcal{I} .

To prove the domain and range axioms, we must show that $\mathcal{G}(\mathcal{I}_d)$ satisfies: $\top \sqsubseteq \forall R^-. \top_i$, $\top \sqsubseteq \forall R. \top_i$, $\top \sqsubseteq \forall R_{ij}^-. \top_i$, $\top \sqsubseteq \forall R_{ij}. \top_j$, where $R \in \mathcal{R}_i$ and R_{ij} 's are the newly introduced role names. We will only prove the first of these in detail; the remaining may be shown similarly. Suppose that $R \in \mathcal{R}_i$ and let $x \in \Delta^{\mathcal{I}} = \bigcup_{i \in V} \Delta^i$. Assume that $y \in \Delta^{\mathcal{I}}$, such that $(x, y) \in R^{\mathcal{I}-}$, i.e., $(y, x) \in R^{\mathcal{I}} = R^i$. Thus, we must have $y \in \Delta^i = \top_i^{\mathcal{I}}$, whence $x \in \{t \in \Delta^{\mathcal{I}} : (\forall y \in \Delta^{\mathcal{I}})((t, y) \in R^{\mathcal{I}} \rightarrow y \in \top_i^{\mathcal{I}})\} = (\forall R^-. \top_i)^{\mathcal{I}}$. This shows that $\top \sqsubseteq \forall R^-. \top_i$ holds in \mathcal{I} .

Finally, suppose that $\#_i(C) \sqsubseteq \#_i(D)$ is in $\mathfrak{R}(\Sigma_d)$. Then $C \sqsubseteq D \in T_i$ and, since $\mathcal{I}_d \models \Sigma_d$, we must have $C^i \subseteq D^i$. By Lemma 17, $\#_i(C)^{\mathcal{I}} \subseteq \#_i(D)^{\mathcal{I}}$, which implies that $\mathcal{I} \models \#_i(C) \sqsubseteq \#_i(D)$. Thus, $\mathcal{G}(\mathcal{I}_d) \models \mathfrak{R}(T_w^*)$. This concludes the proof that, if Σ_d is consistent as witnessed by a package T_w , then \top_w is satisfiable with respect to $\mathfrak{R}(T_w^*)$. \blacksquare

As far as exact completeness is concerned, we have

Theorem 19 (Exact Completeness) *Let $\Sigma_d = \{T_i\}_{i \in V}$ be an F -ALCI ontology. If Σ_d is exactly consistent as witnessed by T_w , then \top_w is satisfiable with respect to $\mathfrak{R}_e(T_w^*)$.*

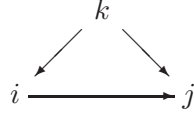
Proof:

Suppose that Σ_d is exactly consistent as witnessed by T_w . Thus, there exists an exact model $\mathcal{I}_d = \langle \{\mathcal{I}_i\}_{i \in V}, \{r_{ij}\}_{(i,j) \in E} \rangle$ of T_w^* , such that $\Delta^w \neq \emptyset$. We proceed to show that $\mathcal{I} := \mathcal{G}(\mathcal{I}_d)$ is a model of $\mathfrak{R}_e(T_w^*)$, such that $\top_w^{\mathcal{I}} \neq \emptyset$.

By the consistency of Σ_d as witnessed by T_w , we have

- $\top_w^{\mathcal{I}} \neq \emptyset$,
- $\mathcal{I} \models C \sqsubseteq \top_i$, for every $C \in \mathcal{C}_i$,
- $\mathcal{I} \models \top \sqsubseteq \forall R^-. \top_i$ and $\mathcal{I} \models \top \sqsubseteq \forall R. \top_i$, for every $R \in \mathcal{R}_i$,
- $\mathcal{I} \models \top \sqsubseteq \forall R_{ij}^-. \top_i$ and $\mathcal{I} \models \top \sqsubseteq \forall R_{ij}. \top_j$, for every $R_{ij}, (i, j) \in E_w$, and that
- $\mathcal{I} \models \#_i(C) \sqsubseteq \#_i(D)$, for every $C \sqsubseteq D \in T_i$, $i \in V_w$.

Thus, it only suffices to show that, for every importing diagram in T_w^* of the form



and all $C \in \overline{\mathcal{C}}_i \cap \overline{\mathcal{C}}_j \cap \overline{\mathcal{C}}_k$, $\mathcal{I} \models \exists R_{ij}^-. (\exists R_{ki}^-. \#_k(C)) = \exists R_{kj}^-. \#_k(C)$. We have

$$\begin{aligned}
 (\exists R_{ij}^-. (\exists R_{ki}^-. \#_k(C)))^{\mathcal{I}} &= R_{ij}^{\mathcal{I}}(R_{ki}^{\mathcal{I}}(\#_k(C)^{\mathcal{I}})) \quad (\text{by Lemma 10}) \\
 &= r_{ij}(r_{ki}(C^k)) \quad (\text{by the definition of } \mathcal{I} \text{ and} \\
 &\quad \text{Lemma 17}) \\
 &= r_{kj}(C^k) \quad (\text{since } \mathcal{I}_d \models^e T_w^*, \text{ see Lemma 6}) \\
 &= R_{kj}^{\mathcal{I}}(\#_k(C)^{\mathcal{I}}) \quad (\text{by the definition of } \mathcal{I} \text{ and} \\
 &\quad \text{Lemma 17}) \\
 &= (\exists R_{kj}^-. \#_k(C))^{\mathcal{I}}. \quad (\text{by Lemma 10})
 \end{aligned}$$

Thus, if $\mathcal{I}_d \models^e T_w^*$, we must have that $\mathfrak{G}(\mathcal{I}_d) \models \mathfrak{R}_e(T_w^*)$. This concludes the proof that, exact consistency of Σ_d with respect to T_w implies that \top_w is satisfiable with respect to $\mathfrak{R}_e(T_w^*)$. ■

1.7 Consequences of Soundness and Completeness

By combining Theorems 13 and 18 we get the following

Theorem 20 (Soundness and Completeness) *Suppose that $\Sigma_d = \{T_i\}_{i \in V}$ is an $F\text{-}\mathcal{ALCI}$ ontology. Σ_d is consistent as witnessed by a module T_w if and only if \top_w is satisfiable with respect to $\mathfrak{R}(T_w^*)$. Moreover, Σ_d is exactly consistent as witnessed by T_w if and only if \top_w is satisfiable with respect to $\mathfrak{R}_e(T_w^*)$.*

By [85] the concept satisfiability, concept subsumption and consistency problems for the language \mathcal{ALC} are PSPACE-complete. By [95], the same problems for the language \mathcal{ALCIQb} are in PSPACE. Thus, since \mathcal{ALC} is a sublanguage of \mathcal{ALCI} , which, in turn, is a sublanguage of \mathcal{ALCIQb} , the aforementioned three problems for the language \mathcal{ALCI} are PSPACE-complete. Since the reductions \mathfrak{R} and \mathfrak{R}_e are obviously doable in polynomial time, we obtain

Theorem 21 *The concept satisfiability, concept subsumption and consistency problems for $F\text{-}\mathcal{ALCI}$ are PSPACE-complete.*

The following theorem, which is a consequence of Theorem 20, shows that a given subsumption is valid as witnessed by a module T_i of an $F\text{-}\mathcal{ALCI}$ ontology T if and only if its translation under $\#_i$ is valid with respect to the reduction $\mathfrak{R}(T_i^*)$. In this case, we say that \mathfrak{R} is a *subsumption-preserving reduction*. Note that this term refers to preservation of subsumptions when passing from a federated ontology to its corresponding single module counterpart and not to a preservation of subsumptions across modules.

Theorem 22 (Subsumption Preservation) *For an $F\text{-}\mathcal{ALCI}$ ontology $\Sigma_d = \{T_i\}_{i \in V}$, it holds that $T_i^* \models_i C \sqsubseteq D$ iff $\mathfrak{R}(T_i^*) \models \#_i(C) \sqsubseteq \#_i(D)$.*

Proof:

Suppose, first, that $T_i^* \models_i C \sqsubseteq D$ and let \mathcal{I} be a model of $\mathfrak{R}(T_i^*)$. Then, by Theorem 13, $F(\mathcal{I})$ is a model of T_i^* , whence, since $T_i^* \models_i C \sqsubseteq D$, we get that $F(\mathcal{I}) \models C \sqsubseteq D$. This implies that $\mathcal{I} \models \#_i(C) \sqsubseteq \#_i(D)$. Therefore $\mathfrak{R}(T_i^*) \models \#_i(C) \sqsubseteq \#_i(D)$.

Conversely, assume that $\mathfrak{R}(T_i^*) \models \#_i(C) \sqsubseteq \#_i(D)$ and let $\mathcal{I}_d \models T_i^*$. Then, by Theorem 18, $G(\mathcal{I}_d) \models \mathfrak{R}(T_i^*)$, whence, since $\mathfrak{R}(T_i^*) \models \#_i(C) \sqsubseteq \#_i(D)$, we get that $G(\mathcal{I}_d) \models \#_i(C) \sqsubseteq \#_i(D)$. This implies that $\mathcal{I}_d \models C \sqsubseteq D$ and, therefore, $T_i^* \models C \sqsubseteq D$. ■

Another consequence of Theorem 20 concerns the *monotonicity* of federated reasoning *with respect to exact models*. More precisely, we show that, given an $F\text{-}\mathcal{ALCI}$ ontology $\Sigma_d = \{T_i\}_{i \in V}$ and an exact model \mathcal{I}_d of Σ_d , a subsumption $C \sqsubseteq D$, with $C, D \in \overline{\mathcal{C}}_i \cap \overline{\mathcal{C}}_j$, $(i, j) \in E$, is valid as witnessed by module T_j provided that it is valid as witnessed by module T_i .

Theorem 23 (Monotonicity) *Let $\Sigma_d = \{T_i\}_{i \in V}$ be an $F\text{-}\mathcal{ALCI}$ ontology and $\mathcal{I}_d = \langle \{\mathcal{I}_i\}_{i \in V}, \{r_{ij}\}_{(i,j) \in E} \rangle$ an exact model of Σ_d . Then, for every $(i, j) \in E$ and $C, D \in \overline{\mathcal{C}}_i \cap \overline{\mathcal{C}}_j$, if $C \sqsubseteq_i D$, then $C \sqsubseteq_j D$.*

Proof:

Suppose that $C \sqsubseteq_i D$. Thus, for every model \mathcal{I} of T_i^* , $C^i \subseteq D^i$. Now consider a model \mathcal{I} of T_j^* . Since $(i, j) \in E$, \mathcal{I} is also a model of T_i^* . Therefore, we have that $C^i \subseteq D^i$, which yields $r_{ij}(C^i) \subseteq r_{ij}(D^i)$. It follows by Exactness (see Definition 8) that $C^j \subseteq D^j$. This proves that $C \sqsubseteq_j D$. ■

It should be stressed that this theorem has a significant limitation. Monotonicity is asserted only for subsumptions that are actually appearing in two different modules of the ontology under consideration. It cannot be asserted for arbitrary subsumptions that may be added later to the ontology. This is due to the fact that, even if the current model is still a model of the augmented federated ontology, it might not be an exact model. Thus, monotonicity is not being applied to arbitrary concept subsumptions in this case, as was done, for instance, in the case of P-DLs. On the other hand the exactness conditions imposed in the present setting are considerably milder than the ones imposed on the P-DL semantics.

In the special case where $D = \perp$, Theorem 23 yields the following corollary:

Corollary 24 (Preservation of Unsatisfiability) *Let $\Sigma_d = \{T_i\}_{i \in V}$ be an $F\text{-}\mathcal{ALCI}$ ontology and $\mathcal{I}_d = \langle \{\mathcal{I}_i\}_{i \in V}, \{r_{ij}\}_{(i,j) \in E} \rangle$ an exact model of Σ_d . Then, for all $(i, j) \in E$ and all $C \in \bar{\mathcal{C}}_i \cap \bar{\mathcal{C}}_j$, if $C \sqsubseteq_i \perp$ then $C \sqsubseteq_j \perp$.*

Thus a concept subsumption $C \sqsubseteq D$, that is unsatisfiable as witnessed by a module T_i , will also be unsatisfiable as witnessed by any other module T_j that imports T_i and shares with T_i the same concepts C, D .

In [8] we listed several desirable properties that modular ontologies may satisfy and imposed various conditions on the interpretations of P-DLs to enforce these desiderata. The list included preservation of unsatisfiability, transitive reusability of knowledge (which is a consequence of monotonicity) and contextualized interpretation of knowledge. In the present work, we chose to consider arbitrary models that do not necessarily satisfy either monotonicity of reasoning or preservation of unsatisfiability. Contextualization of knowledge is satisfied by default. By restricting to exact interpretations, the first two properties in the list reemerge.

1.8 Summary

In this paper we have introduced a modular ontology language, contextualized federated description logic $F\text{-}\mathcal{ALCI}$, that allows reuse of knowledge from multiple ontologies. An $F\text{-}\mathcal{ALCI}$ ontology consists of multiple ontology modules each of which can be viewed as an \mathcal{ALCI} ontology. Concept and role names can be shared by “importing” relations among modules.

The proposed language supports contextualized interpretations, i.e., interpretations from the point of view of a specific module. We have insisted on very loose constraints on image domain relations, i.e., the relations between individuals in different local domains, while still retaining harmonious coordination between the local ontology modules. However, if additional properties are desired, such as the preservation of satisfiability of concept expressions, the monotonicity of inference, or the transitive reusability of knowledge, then more restrictive conditions have to be imposed on the proposed semantics. We have shown how this can be achieved in a case of particular interest.

Ongoing work is aimed at developing a distributed reasoning algorithm for $F\text{-}\mathcal{ALCI}$ by extending the results of [4, 6] and [73].

Chapter 2. A FEDERATED TABLEAU ALGORITHM FOR F- \mathcal{ALCI}

Abstract

Many semantic web applications require support for knowledge representation and inference over a federation of multiple autonomous ontology modules, without having to combine them in one location. Federated \mathcal{ALCI} or F- \mathcal{ALCI} is a modular description logic, each of whose modules is roughly an \mathcal{ALCI} ontology (\mathcal{ALC} with inverse roles). F- \mathcal{ALCI} supports importing of both concepts and roles across modules as well as contextualized interpretation of logical connectives. We present a federated tableau algorithm for reasoning with a collection of interlinked F- \mathcal{ALCI} ontology modules without the need to combine the modules into a single ontology. Local reasoners apply tableau expansion rules as in the ordinary \mathcal{ALCI} tableau algorithm. Coordination is achieved by message exchanges between local tableaux maintained by the individual reasoners. We prove soundness and completeness of the federated tableau algorithm and show that its worst-case running time is nondeterministic doubly exponential in the size of the largest ontology module.

2.1 Introduction

In its traditional form the world-wide web consists of data that is geared towards human understanding and processing. However, the rapid increase in the amount and complexity of information available on the web calls for methods for its automated analysis and interpretation. Thus, there is an increasing emphasis on machine interpretable representations of information with the goal of transforming the current web into a semantic web. The most common way to-date to represent and reason about information on the semantic web is by organizing it into various ontologies, or clusters of domain-specific data. Each ontology ad-

dresses a particular domain of knowledge. Ontologies are usually developed and maintained independently by autonomous groups that borrow terminology, facts, instances etc, from each other. They are usually built using ontology languages, such as OWL [50]. OWL is now recommended by the W3C consortium [9]. Ontology languages are based on Description Logics (DLs), which constitute a family of logic-based knowledge representation languages. They are ordinarily decidable fragments of first-order logic or decidable extensions of those fragments. They are often equipped with tableau-based decision procedures for problems such as computing subsumption hierarchies of concepts, satisfiability of concepts, or instances of a concept expression. In [51] a decision procedure for the very expressive DL *SHOIQ* is presented. It extends a tableau algorithm for *SHIQ* [52], which gave rise to several implemented reasoners [46, 76, 87].

The need to balance the requirement of autonomy against those of collaboration in developing ontologies has recently led to increasing interest in modular ontologies. These are ontologies that are physically and/or conceptually distributed. Each module of such an ontology is independently developed to address specific aspects or subdomain of expertise of a large domain of knowledge. The modules are interdependent in the sense that the various subdomains are most conveniently described by borrowing concepts and data from each other. This supports the autonomy of different groups engaged in developing ontology modules in their respective areas of expertise, eliminates duplication and redundancy and encourages modularity in the construction of ontologies. Multiple modular ontology languages have been proposed to facilitate such an autonomous collaborative development of ontologies. Among them, the best known are Distributed Description Logics (DDL) [14], \mathcal{E} -Connections [26, 56] and Package-Based Description Logics (P-DLs) [8].

Since its inception, the semantic web has been envisioned as a non-centralized, highly distributed collection of ontologies with a degree of redundant and overlapping knowledge [11]. The decentralized nature of the web necessarily implies that distant and diverse users will create and use their own local ontologies to organize and reason about their data depending on their special needs. The independence and autonomy clearly makes it easier for these local

communities to revise, update, or modify their ontologies according to their own requirements or to accommodate new data. This scenario calls for a distributed reasoning approach in which ontologies have each its own reasoning capabilities and the reasoners communicate with each other whenever a need arises. Computationally, it is advantageous to do as much reasoning as possible at the local level, taking advantage of the specific properties of the local environment. On many occasions it is not even feasible to reason with a “centralized” ontology resulting from integrating the various modules because its size is too large for such an integration to be efficient. In other cases, such an integration may not be possible because the autonomous ontology modules can only selectively share information with each other due to issues of security, privacy, copyright etc. These factors accentuate the need for a federated approach to reasoning that does not require the physical integration of the ontology modules. Numerous researchers, including Serafini et al. [81, 83, 82], have argued eloquently in favor of the modular approach and even designed specific systems/architectures (DRAGO) to facilitate decentralized reasoning over distributed ontologies. Some authors advocated partitioning of large ontologies into smaller but computationally “leaner & meaner” and possibly more coherent ontologies [94, 80]. Against this background, several algorithms have been presented for the three paradigms mentioned above. Serafini et al. [81, 83, 82] introduced a tableau algorithm for reasoning with DDL. Grau et al. [28, 27] present a tableau procedure for \mathcal{E} -Connections. Finally, Bao et al. [4] present distributed reasoning algorithms for P-DLs. As contrasted to this work, the approach of [25] emphasizes privacy and deals with situations where open sharing of data is not allowed, whereas [101] deals with dynamic, temporal and intensional aspects of knowledge.

The main goal of this paper is to present a federated tableau-based algorithm for the fully contextualized federated description logic $F\text{-}\mathcal{ALCI}$, introduced in [97]. Each of the $F\text{-}\mathcal{ALCI}$ modules is an \mathcal{ALCI} ontology. One of the distinctive features of this modular language is that it contextualizes all logical connectives (contrast with P-DLs, where only negation is contextual). Moreover, it allows greater semantic flexibility than P-DLs at the expense of some properties that may be desirable in some contexts, but not in others, such as transitive reusability of knowledge and preservation of concept unsatisfiability. We present a nondeterministic doubly

exponential federated tableau-based algorithm, that allows us to test concept satisfiability in F- \mathcal{ALCI} from a specific module's point of view. Among its novel features, specifically designed to handle contextualized connectives, are: (a) a new normal form for concept expressions, called negation local form, replacing negation normal form, (b) new “contextual” tableaux expansion rules and (c) a specially tailored synchronization mechanism based on message exchanges. Although it is well-known (see [97]) that a F- \mathcal{ALCI} ontology can be integrated in a context sensitive way into a single \mathcal{ALCI} ontology, the algorithm presented here does not require such an integration.

2.2 F- \mathcal{ALCI} Syntax and Semantics

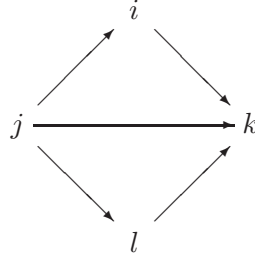
Let $G = \langle V, E \rangle$, with $V = \{1, 2, \dots, n\}$, be a directed acyclic graph augmented with loops. The loops offer technical convenience. For instance, if we say that a j -concept name is also an i -concept, if $(j, i) \in E$, this will include the case $j = i$ (see Definition (2.1) below). The nodes of this graph represent modules of the federated ontology and the edges represent, roughly speaking, direct importing relations, i.e., allowable importing links through which a target module may import either concepts or roles from the source module. For every node $i \in V$, the i -language always includes a set \mathcal{C}_i of *i-concept names* and a set \mathcal{R}_i of *i-role names*. The set $\widehat{\mathcal{R}}_i$ of *i-role expressions* consists of expressions of the form R, R^- , with $R \in \mathcal{R}_j, (j, i) \in E$ (R^- stands for the inverse of R). The set $\widehat{\mathcal{C}}_i$ of *i-concept expressions* consists of recursively defined expressions of the form:

$$C \in \mathcal{C}_j, \top_j, \perp_j, \neg_j C, C \sqcap_j D, C \sqcup_j D, \exists_j R.C, \forall_j R.C, (j, i) \in E, \quad (2.1)$$

where $C, D \in \widehat{\mathcal{C}}_i \cap \widehat{\mathcal{C}}_j$ and $R \in \widehat{\mathcal{R}}_i \cap \widehat{\mathcal{R}}_j$. The *i-formulas* are of the form $C \sqsubseteq D$, with $C, D \in \widehat{\mathcal{C}}_i$. A **local TBox** T_i is a finite set of *i-formulas* and a **knowledge base** (KB) or **TBox** is a collection $T = \{T_i\}_{i \in V}$.

Example 1: Consider a F- \mathcal{ALCI} ontology with four modules i, j, k, l and importing relations

modeled by the following graph G (not showing self-loops):



Let $D \in \mathcal{C}_i, E, F \in \mathcal{C}_j$ and $G \in \mathcal{C}_l$. Then, as the reader may easily check, we have $\neg_k((D \sqcap_i \neg_j(E \sqcup_j F)) \sqcup_k \neg_k(E \sqcap_l G)) \in \widehat{\mathcal{C}}_k$. \square

We turn now to the semantics of F- \mathcal{ALCI} . An **interpretation** $\mathcal{I} = \langle \{\mathcal{I}_i\}_{i \in V}, \{r_{ij}\}_{(i,j) \in E} \rangle$ consists of a family $\mathcal{I}_i = \langle \Delta^i, \cdot^i \rangle, i \in V$, of **local interpretations**, together with a family of **image domain relations** $r_{ij} \subseteq \Delta^i \times \Delta^j, (i,j) \in E$, such that $r_{ii} = \text{id}_{\Delta^i}$, for all $i \in V$. We require that at least one of the local domains Δ^i be nonempty. For a binary relation $r \subseteq \Delta^i \times \Delta^j, X \subseteq \Delta^i$ and $S \subseteq \Delta^i \times \Delta^i$, we set

$$r(X) := \{y \in \Delta^j : (\exists x \in X)((x, y) \in r)\},$$

$$r(S) := \{(z, w) \in \Delta^j \times \Delta^j : (\exists (x, y) \in S)((x, z), (y, w) \in r)\}$$

to denote the images of X and S under the binary relation r . If $X = \{x\}$, we write $r(x)$ instead of $r(\{x\})$ and we follow similar simplifying conventions when confusion is unlikely. The basic features of the local interpretation function \cdot^i are as follows (see [97]):

- $C^i \subseteq \Delta^i$, for all $C \in \mathcal{C}_i$,
- $C^i = r_{ji}(C^j)$, for all $(j, i) \in E$ and $C \in \mathcal{C}_j \cap \widehat{\mathcal{C}}_i$,
- $R^i \subseteq \Delta^i \times \Delta^i$, for all $R \in \mathcal{R}_i$,
- $R^i = r_{ji}(R^j)$, for all $R \in \mathcal{R}_j \cap \widehat{\mathcal{R}}_i$,
- $\top_j^i = r_{ji}(\Delta^j)$, $\perp_j^i = \emptyset$.

The recursive features of the local interpretation function \cdot^i are as follows, for all $R \in \widehat{\mathcal{R}}_i$ and $C, D \in \widehat{\mathcal{C}}_i$:

- $R^{-i} = R^{i-}$
- $(\neg_j C)^i = r_{ji}(\Delta^j \setminus C^j)$
- $(C \sqcap_j D)^i = r_{ji}(C^j \cap D^j)$
- $(C \sqcup_j D)^i = r_{ji}(C^j \cup D^j)$
- $(\exists_j R.C)^i = r_{ji}(\{x \in \Delta^j : (\exists y)((x, y) \in R^j \text{ and } y \in C^j)\})$
- $(\forall_j R.C)^i = r_{ji}(\{x \in \Delta^j : (\forall y)((x, y) \in R^j \text{ implies } y \in C^j)\})$

For all $i \in V$, **i -satisfiability**, denoted by \models_i , is defined by $\mathcal{I} \models_i C \sqsubseteq D$ iff $C^i \subseteq D^i$. Given a TBox $T = \{T_i\}_{i \in V}$, $\mathcal{I} \models_i T_i$ iff $\mathcal{I} \models_i \tau$, for every $\tau \in T_i$. $\mathcal{I} \models T$ iff $\mathcal{I} \models_i T_i$, for every $i \in V$. An interpretation $\mathcal{I} = \langle \{\mathcal{I}_i\}_{i \in V}, \{r_{ij}\}_{(i,j) \in E} \rangle$ is a **model** of a F- \mathcal{ALCI} KB $T = \{T_i\}_{i \in V}$ if $\mathcal{I} \models T$.

Given a node $w \in V$, let $G_w = \langle V_w, E_w \rangle$ be the subgraph of G induced by the subset of vertices of G from which the vertex w is reachable. Given a KB $T = \{T_i\}_{i \in V}$, let $T_w^* = \{T_i\}_{i \in V_w}$ be the **importing closure** of w . T is **consistent as witnessed by a module** T_w if T_w^* has a model $\mathcal{I} = \langle \{\mathcal{I}_i\}_{i \in V_w}, \{r_{ij}\}_{(i,j) \in E_w} \rangle$, such that $\Delta^w \neq \emptyset$. A concept C is **satisfiable as witnessed by** T_w if there is a model of T_w^* , such that $C^w \neq \emptyset$. A concept subsumption $C \sqsubseteq D$ is **valid as witnessed by** T_w , denoted by $C \sqsubseteq_w D$, if for every model of T_w^* , $C^w \subseteq D^w$. We use $C \equiv_w D$ as the abbreviation of $C \sqsubseteq_w D$ and $D \sqsubseteq_w C$. It becomes clear from these definitions that in F- \mathcal{ALCI} the consistency, satisfiability and subsumption problems are always answered from the local point of view of a *witness module*. Furthermore, it is possible for different modules to draw different conclusions from their own points of view.

2.3 Negation Local Form of Concept Expressions

Before introducing the notion of tableau for F- \mathcal{ALCI} , we will discuss a special normal form that we need in place of the negation normal form, which does not seem to exist for F- \mathcal{ALCI} -concept expressions. The need arises from the fact that, in most tableaux algorithms

for description logics, the input is first transformed into negation normal form, i.e., a form in which negation occurs only before concept names. To illustrate why the transformation to negation normal form is problematic in the case of contextualized connectives, consider the following example:

Example 2: Let T_1 and T_2 be two modules and assume that A and B are concept names in \mathcal{C}_1 and that T_2 is allowed to import names and connectives from T_1 . Consider the extension

$$(\neg_2(A \sqcap_1 B))^2 = \Delta^2 \setminus r_{12}(A^1 \cap B^1).$$

Note that the expression $\neg_2 A \sqcup_1 \neg_2 B$ does not even make sense because T_1 is not allowed to import concepts and connectives from T_2 . So the only hope for a negation normal form for the concept expression $\neg_2(A \sqcap_1 B)$ in T_2 would be $\neg_1 A \sqcup_1 \neg_1 B$. But its extension is

$$(\neg_1 A \sqcup_1 \neg_1 B)^2 = r_{12}(\Delta^1 \setminus (A^1 \cap B^1)),$$

which, since r_{12} is an arbitrary relation, is not guaranteed to equal $\Delta^2 \setminus r_{12}(A^1 \cap B^1)$. This example unveils some of the difficulties encountered when one attempts to discover a possible normal form for concept expressions that deals with negation and preserves the relevant semantics. \square

In the present context, it will be assumed that all concepts are in a variant of the negation normal form, which will be called **negation local form** (NLF). The transformation to NLF affects only concept expressions containing i -negations appearing in module T_i before other i -connectives. In this case, the i -negation is pushed “inward” using a number of simple syntactical rules, similar to the ones used to transform an ordinary \mathcal{ALC} -formula into negation normal form. The NLF of an i -concept $C \in \widehat{\mathcal{C}}_i$ is denoted by $\mathbf{nlf}_i(C)$. It is defined recursively on the structure of concepts in $\widehat{\mathcal{C}}_i$ by applying the following rules:

- $\mathbf{nlf}_i(\top_j) = \top_j, \mathbf{nlf}_i(\perp_j) = \perp_j$

- $\mathbf{nlf}_i(\neg_j \top_k) = \begin{cases} \perp_i, & \text{if } j = k = i, \\ \neg_j \top_k, & \text{otherwise} \end{cases}, \mathbf{nlf}_i(\neg_j \perp_k) = \begin{cases} \top_i, & \text{if } j = k = i, \\ \neg_j \perp_k, & \text{otherwise} \end{cases}, \text{ for all } (j, i) \in E$
- $\mathbf{nlf}_i(C) = C$, for all $(j, i) \in E$ and all $C \in \mathcal{C}_j$;
- - $\mathbf{nlf}_i(\neg_j C) = \neg_j C$, for all $(j, i) \in E$ and all $C \in \mathcal{C}_k \cap \widehat{\mathcal{C}}_i \cap \widehat{\mathcal{C}}_j$;
 - $\mathbf{nlf}_i(\neg_j \neg_k C) = \begin{cases} \mathbf{nlf}_i(C), & \text{if } j = k = i \\ \neg_j \neg_k \mathbf{nlf}_i(C), & \text{otherwise} \end{cases}, (j, i) \in E, \neg_k C \in \widehat{\mathcal{C}}_i \cap \widehat{\mathcal{C}}_j$;
 - $\mathbf{nlf}_i(\neg_j (C \sqcap_k D)) = \begin{cases} \neg_i \mathbf{nlf}_i(C) \sqcup_i \neg_i \mathbf{nlf}_i(D), & \text{if } i = j = k \\ \neg_j (\mathbf{nlf}_i(C) \sqcap_k \mathbf{nlf}_i(D)), & \text{otherwise} \end{cases}, \text{ for all } (j, i) \in E, C, D \in \widehat{\mathcal{C}}_i \cap \widehat{\mathcal{C}}_j$;
 - $\mathbf{nlf}_i(\neg_j (C \sqcup_k D))$ is similar;
 - $\mathbf{nlf}_i(\neg_j \forall_k R.C) = \begin{cases} \exists_i R. \neg_i \mathbf{nlf}_i(C), & \text{if } i = j = k \\ \neg_j \forall_k R. \mathbf{nlf}_i(C), & \text{otherwise} \end{cases}, \text{ for all } (j, i) \in E \text{ and } \forall_k R.C \in \widehat{\mathcal{C}}_i \cap \widehat{\mathcal{C}}_j$;
 - $\mathbf{nlf}_i(\neg_j \exists_k R.C)$ is similar;
- $\mathbf{nlf}_i(C \sqcap_j D) = \mathbf{nlf}_j(C) \sqcap_j \mathbf{nlf}_j(D)$, for all $(j, i) \in E$;
- $\mathbf{nlf}_i(C \sqcup_j D) = \mathbf{nlf}_j(C) \sqcup_j \mathbf{nlf}_j(D)$, for all $(j, i) \in E$;
- $\mathbf{nlf}_i(\forall_j R.C) = \forall_j R. \mathbf{nlf}_j(C)$, for all $(j, i) \in E$;
- $\mathbf{nlf}_i(\exists_j R.C) = \exists_j R. \mathbf{nlf}_j(C)$, for all $(j, i) \in E$.

Example 3: Consider the k -concept expression

$$C = \neg_k((D \sqcap_i \neg_j(E \sqcup_j F)) \sqcup_k \neg_k(E \sqcap_l G)),$$

where D, E, F and G are concept names, i, j and k are distinct and the concept expression is valid, i.e., the importing relations that allow building this concept expression in $\widehat{\mathcal{C}}_k$ hold (see

Example 1). If we apply \mathbf{nlf}_k to it, we get

$$\begin{aligned}
\mathbf{nlf}_k(C) &= \neg_k \mathbf{nlf}_k(D \sqcap_i \neg_j(E \sqcup_j F)) \sqcap_k \neg_k \mathbf{nlf}_k(\neg_k(E \sqcap_l G)) \\
&= \neg_k(\mathbf{nlf}_k(D) \sqcap_i \mathbf{nlf}_k(\neg_j(E \sqcup_j F))) \sqcap_k \neg_k \neg_k(\mathbf{nlf}_k(E) \sqcap_l \mathbf{nlf}_k(G)) \\
&= \neg_k(D \sqcap_i (\neg_j(\mathbf{nlf}_k(E) \sqcup_j \mathbf{nlf}_k(F)))) \sqcap_k (E \sqcap_l G) \\
&= \neg_k(D \sqcap_i (\neg_j(E \sqcup_j F))) \sqcap_k (E \sqcap_l G)
\end{aligned}$$

□

The next lemma asserts that the transformation from a concept $C \in \widehat{\mathcal{C}}_i$ into its negation local form $\mathbf{nlf}_i(C)$ does not change its meaning from the point of view of module T_i .

Lemma 25 *Let $\Sigma = \{T_i\}_{i \in V}$ be a $F\text{-}\mathcal{ALCC}$ KB, $\mathcal{I} = \langle \{\mathcal{I}_i\}_{i \in V}, \{r_{ij}\}_{(i,j) \in E} \rangle$ an interpretation for Σ , $i \in V$ and $C \in \widehat{\mathcal{C}}_i$. Then $(\mathbf{nlf}_i(C))^i = C^i$.*

Proof:

We employ structural induction on C .

Suppose, first, that $(j, i) \in E$ and $C \in \mathcal{C}_j$. Then $(\mathbf{nlf}_i(C))^i = C^i$ by the definition of $\mathbf{nlf}_i(C)$. Similarly $(\mathbf{nlf}_i(\top_j))^i = \top_j^i$ and $(\mathbf{nlf}_i(\perp_j))^i = \perp_j^i$.

For $C \sqcap_j D$, we have

$$\begin{aligned}
(\mathbf{nlf}_i(C \sqcap_j D))^i &= (\mathbf{nlf}_j(C) \sqcap_j \mathbf{nlf}_j(D))^i \quad (\text{by the definition of } \mathbf{nlf}_i) \\
&= r_{ji}((\mathbf{nlf}_j(C))^j \cap (\mathbf{nlf}_j(D))^j) \quad (\text{by the definition of } \cdot^i) \\
&= r_{ji}(C^j \cap D^j) \quad (\text{by the induction hypothesis}) \\
&= (C \sqcap_j D)^i. \quad (\text{by the definition of } \cdot^i)
\end{aligned}$$

The case of $C \sqcup_j D$ may be handled similarly.

For $\exists_j R.C$, we get

$$\begin{aligned}
(\mathbf{nlf}_i(\exists_j R.C))^i &= (\exists_j R.\mathbf{nlf}_j(C))^i \quad (\text{by the definition of } \mathbf{nlf}_i) \\
&= r_{ji}(\{x \in \Delta^j : (\exists y \in (\mathbf{nlf}_j(C))^j)((x, y) \in R^j)\}) \\
&\quad (\text{by the definition of } \cdot^i) \\
&= r_{ji}(\{x \in \Delta^j : (\exists y \in C^j)((x, y) \in R^j)\}) \\
&\quad (\text{by the induction hypothesis}) \\
&= (\exists_j R.C)^i. \quad (\text{by the definition of } \cdot^i)
\end{aligned}$$

The case of $\forall_j R.C$ may be handled similarly.

Next, we turn to negation and concentrate on the forms that do change. We have

- $(\mathbf{nlf}_i(\neg_i \top_i))^i = \perp_i^i = \emptyset = \Delta^i \setminus \Delta^i = \Delta^i \setminus \top_i^i = (\neg_i \top_i)^i.$
- $(\mathbf{nlf}_i(\neg_i \perp_i))^i = \top_i^i = \Delta^i = \Delta^i \setminus \emptyset = \Delta^i \setminus \perp_i^i = (\neg_i \perp_i)^i.$
- $(\mathbf{nlf}_i(\neg_i \neg_i C))^i = C^i = \Delta^i \setminus (\Delta^i \setminus C^i) = (\neg_i \neg_i C)^i.$
- For $\neg_i(C \sqcap_i D)$ we have

$$\begin{aligned}
(\mathbf{nlf}_i(\neg_i(C \sqcap_i D)))^i &= (\neg_i \mathbf{nlf}_i(C) \sqcup_i \neg_i \mathbf{nlf}_i(D))^i \\
&= (\Delta^i \setminus \mathbf{nlf}_i(C)^i) \cup (\Delta^i \setminus \mathbf{nlf}_i(D)^i) \\
&= (\Delta^i \setminus C^i) \cup (\Delta^i \setminus D^i) \\
&= \Delta^i \setminus (C^i \cap D^i) \\
&= (\neg_i(C \sqcap_i D))^i.
\end{aligned}$$

- The case of $\neg_j(C \sqcup_k D)$ is handled similarly.

- For $\neg_j \exists_k R.C$ we have

$$\begin{aligned}
(\mathbf{nlf}_i(\neg_i \exists_i R.C))^i &= (\forall_i R. \neg_i C)^i \\
&= \{x \in \Delta^i : (\forall y)((x, y) \in R^i \rightarrow y \notin C^i)\} \\
&= \Delta^i \setminus \{x \in \Delta^i : (\exists y)((x, y) \in R^i \text{ and } y \in C^i)\} \\
&= \Delta^i \setminus (\exists_i R.C)^i \\
&= (\neg_i \exists_i R.C)^i.
\end{aligned}$$

- Finally, the case of $\neg_j \forall_k R.C$ is handled similarly.

■

As far as the NLF is concerned, the reader should notice that, in module T_i , an i -negation is either followed by a concept name $C \in \mathcal{C}_i$ or by a concept name $C \in \mathcal{C}_j \cap \widehat{\mathcal{C}}_i$, $(j, i) \in E$, $j \neq i$, or by a concept C , whose outermost connective is a j -connective, for some $j \neq i$. The first kind will be called **of type 1** and the remaining two kinds **of type 2 with trace j** . This observation will be important for the formulation and analysis of the distributive algorithm and will be called upon many times in the sequel.

2.4 Federated Tableaux for F- \mathcal{ALCI}

Tableau-based algorithms are used to test satisfiability of concepts in description logics. The main idea behind the F- \mathcal{ALCI} tableau algorithm is to construct multiple, federated local tableaux, one for each module, using, to the furthest extent possible, only knowledge locally available to that module. The coordination between local tableaux is achieved via inter-module messages which relate pairs of elements across different local tableaux. In effect, this will build a representation of possible image-domain relations r_{ij} , for $(i, j) \in E$. An i -subconcept of an i -concept expression C is a substring of C , which forms also an i -concept expression. We make this notion precise in Definition 26. It will be used in the definition of a federated tableau for a F- \mathcal{ALCI} -concept in NLF with respect to a module T_w , that follows.

Definition 26 The i -subconcepts $\text{sub}_i(C)$ of an $F\text{-}\mathcal{ALCI}$ concept $C \in \widehat{\mathcal{C}}_i$ in NLF is inductively defined as:

$$\begin{aligned} \text{sub}_i(A) &= \{A\}, \quad A \in \bigcup_{(j,i) \in E} (\mathcal{C}_j \cup \{\top_j, \perp_j\}) \\ \text{sub}_i(C \boxplus_j D) &= \{C \boxplus_j D\} \cup \text{sub}_j(C) \cup \text{sub}_j(D), \quad \boxplus \in \{\sqcap, \sqcup\}, (j,i) \in E, \\ \text{sub}_i(\bowtie_j R.C) &= \{\bowtie_j R.C\} \cup \text{sub}_j(C), \quad \bowtie \in \{\exists, \forall\}, (j,i) \in E, \\ \text{sub}_i(\neg_j C) &= \{\neg_j C\} \cup \text{sub}_j(C), \quad (j,i) \in E, j \neq i, \\ \text{sub}_i(\neg_i C) &= \begin{cases} \{\neg_i C\} \cup \text{sub}_i(C), & \text{if } \neg_i C \text{ is of type 1} \\ \{\neg_i C\} \cup \text{sub}_j(C), & \text{if } \neg_i C \text{ is of type 2 with trace } j \end{cases} \end{aligned}$$

Moreover, define, for every concept expression $C \in \widehat{\mathcal{C}}_i$, $\text{Rol}(C) \subseteq \widehat{\mathcal{R}}_i$ to be the (finite) set of role expressions appearing in C .

For every module T_i , we define

$$C_{T_i} = \top_i \sqcap_i \prod_{C \sqsubseteq D \in T_i} (\mathbf{nlf}_i(\neg_i C) \sqcup_i \mathbf{nlf}_i(D)),$$

where the \prod also refers to the i -th conjunction symbol.

Let T_w be a module and $D \in \widehat{\mathcal{C}}_w$ an $F\text{-}\mathcal{ALCI}$ concept in NLF. A **federated tableau for D with respect to T_w** is a tuple $M = \langle \{M_i\}_{i \in V_w}, \{m_{ij}\}_{(i,j) \in E_w} \rangle$, where each M_i is a local tableau, for $i \in V_w$, and m_{ij} is a **tableau relation** from a local tableau M_i to a local tableau M_j , for $(i,j) \in E_w$.

Each local tableau is a tuple $M_i = \langle U_i, F_i, \mathcal{L}_i \rangle$, where

- U_i is a set of individuals,
- $F_i \subseteq U_i \times U_i$ is a binary relation on U_i ,
- \mathcal{L}_w is a label function that assigns elements of $2^{\text{sub}_w(D) \cup \text{sub}_w(C_{T_w})}$ to individuals in U_w and elements of $2^{\text{Rol}(D) \cup \text{Rol}(C_{T_w})}$ to pairs in F_w whereas \mathcal{L}_i is a label function that assigns elements of $2^{\text{sub}_i(C_{T_i})}$ to individuals in U_i and elements of $2^{\text{Rol}(C_{T_i})}$ to pairs in F_i , for $i \neq w$.

Each tableau relation m_{ij} is a subset of $U_i \times U_j$, $(i, j) \in E_w$.

The federated tableau M should satisfy the following conditions:

- (D1) there exists $x \in U_w$, such that $D \in \mathcal{L}_w(x)$;
- (D2) for every $x \in U_i$, $C_{T_i} \in \mathcal{L}_i(x)$;
- (B1) $C \in \mathcal{L}_i(x)$ iff there exists $x' \in U_j$, with $(x', x) \in m_{ji}$, such that $C \in \mathcal{L}_j(x')$, for all $C \in \widehat{\mathcal{C}}_i \cap (\mathcal{C}_j \cup \{\top_j\})$, $(j, i) \in E_w$;
- (B2) $R \in \mathcal{L}_i(\langle x, y \rangle)$ iff there exist $x', y' \in U_j$, with $(x', x), (y', y) \in m_{ji}$, such that $R \in \mathcal{L}_j(\langle x', y' \rangle)$, for all $R \in \widehat{\mathcal{R}}_i \cap \mathcal{R}_j$, $(j, i) \in E_w$;
- (N1) if $C \in \mathcal{L}_i(x)$, then $\neg_i C \notin \mathcal{L}_i(x)$, for every $C \in \mathcal{C}_i$;
- (N2) if $\neg_i C \in \mathcal{L}_i(x)$ is of type 2 with trace j , then, if $x' \in U_j$, with $(x', x) \in m_{ji}$, then $\neg_j C \in \mathcal{L}_j(x')$, for all $C \in \widehat{\mathcal{C}}_i \cap \widehat{\mathcal{C}}_j$, $(j, i) \in E_w, j \neq i$;
- (N3) if $\neg_j C \in \mathcal{L}_i(x)$, then, there exists $x' \in U_j$, with $(x', x) \in m_{ji}$ and $\neg_j C \in \mathcal{L}_j(x')$, for all $C \in \widehat{\mathcal{C}}_i \cap \widehat{\mathcal{C}}_j$, $(j, i) \in E_w, j \neq i$;
- (A1) if $C_1 \sqcap_j C_2 \in \mathcal{L}_i(x)$, then, if $i = j$, $C_1, C_2 \in \mathcal{L}_i(x)$ and, if $i \neq j$, then, there exists $x' \in U_j$, with $(x', x) \in m_{ji}$, such that $C_1 \sqcap_j C_2 \in \mathcal{L}_j(x')$, for every $C_1, C_2 \in \widehat{\mathcal{C}}_i \cap \widehat{\mathcal{C}}_j$ and $(j, i) \in E_w$;
- (A2) if $C_1 \sqcup_j C_2 \in \mathcal{L}_i(x)$, then, if $i = j$, then $C_1 \in \mathcal{L}_i(x)$ or $C_2 \in \mathcal{L}_i(x)$ and, if $i \neq j$, then, there exists $x' \in U_j$, with $(x', x) \in m_{ji}$, such that $C_1 \sqcup_j C_2 \in \mathcal{L}_j(x')$, for every $C_1, C_2 \in \widehat{\mathcal{C}}_i \cap \widehat{\mathcal{C}}_j$ and $(j, i) \in E_w$;
- (A3) if $\forall_j R.C \in \mathcal{L}_i(x)$, then, if $i = j$, then, for all $y \in U_i$, such that $R \in \mathcal{L}_i(\langle x, y \rangle)$, we have $C \in \mathcal{L}_i(y)$, and, if $i \neq j$, then, there exists $x' \in U_j$, with $(x', x) \in m_{ji}$, such that $\forall_j R.C \in \mathcal{L}_j(x')$, for all $R \in \widehat{\mathcal{R}}_i \cap \widehat{\mathcal{R}}_j, C \in \widehat{\mathcal{C}}_i \cap \widehat{\mathcal{C}}_j, (j, i) \in E_w$;
- (A4) if $\exists_j R.C \in \mathcal{L}_i(x)$, then, if $i = j$, then, there exists $y \in U_i$, such that $R \in \mathcal{L}_i(\langle x, y \rangle)$ and $C \in \mathcal{L}_i(y)$, and, if $i \neq j$, then, there exists $x' \in U_j$, with $(x', x) \in m_{ji}$, such that $\exists_j R.C \in \mathcal{L}_j(x')$, for all $R \in \widehat{\mathcal{R}}_i \cap \widehat{\mathcal{R}}_j, C \in \widehat{\mathcal{C}}_i \cap \widehat{\mathcal{C}}_j, (j, i) \in E_w$.

Condition (D1) ensures that the interpretation of D in the model described by the federated tableau is nonempty. Condition (D2) ensures the satisfiability of all federated TBox axioms in the model. Conditions (B1) and (B2) stipulate that the interpretations of imported concept names and imported role names are inherited from their corresponding interpretations in their original module. Conditions (N1)-(N3) guarantee that all relevant properties of the contextualized negations will be satisfied in the resulting model. In particular, Conditions (N1) and (N2) safeguard the consistency of the model. Conditions (A1)-(A4) ensure the correctness of the interpretation of the remaining localized connectives.

The following two lemmas establish the correspondence between concept satisfiability, and, thus, also between TBox consistency and concept subsumption, and the existence of a federated tableau for that concept in $F\text{-}\mathcal{ALCI}$.

Lemma 27 *Let $T = \{T_i\}_{i \in V}$ be a $F\text{-}\mathcal{ALCI}$ KB and D be concept in T_w . If D has a federated tableau w.r.t. T_w , then D is satisfiable as witnessed by T_w .*

Proof:

Let $\langle \{M_i\}_{i \in V_w}, \{m_{ij}\}_{(i,j) \in E_w} \rangle$, with $M_i = \langle U_i, F_i, \mathcal{L}_i \rangle$, be a tableau for D w.r.t. T_w^* . Then, a federated model $\mathcal{I} = \langle \{\mathcal{I}_i\}_{i \in V_w}, \{r_{ij}\}_{(i,j) \in E_w} \rangle$ of T_w^* may be defined as follows:

$$\begin{aligned} \Delta^i &= U_i; \\ A^i &= \{x \in U_i : A \in \mathcal{L}_i(x)\}, \text{ for every } i\text{-concept name } A; \\ R^i &= \{\langle x, y \rangle \in F_i : R \in \mathcal{L}_i(\langle x, y \rangle)\}, \text{ for every } i\text{-role name } R; \\ r_{ij} &= m_{ij}. \end{aligned}$$

By using induction on the structure of an i -concept, we show that

$$C \in \mathcal{L}_i(x) \quad \text{implies} \quad x \in C^i. \quad (2.2)$$

- If C is an i -concept name, then $C \in \mathcal{L}_i(x)$ if and only if, by the definition of C^i , $x \in C^i$.

- If C is a j -concept name or \top_j , $j \neq i$, and $C \in \mathcal{L}_i(x)$, then, by Property (B1), there exists $x' \in U_j$, with $(x', x) \in m_{ji} = r_{ji}$, such that $C \in \mathcal{L}_j(x')$. Therefore $x \in r_{ji}(x') \subseteq r_{ji}(C^j) = C^i$.
- Suppose that $\neg_i C \in \mathcal{L}_i(x)$ is of type 1. Then, by Property (N1), $C \notin \mathcal{L}_i(x)$, whence $x \notin C^i$, i.e., $x \in (\neg_i C)^i$.
- Suppose that $\neg_i C \in \mathcal{L}_i(x)$ is of type 2 with trace j . We must show that $x \in (\neg_i C)^i = \Delta^i \setminus C^i = \Delta^i \setminus r_{ji}(C^j)$. Suppose, to the contrary, that $x \in r_{ji}(C^j)$. Then, there exists $x' \in U_j$, with $(x', x) \in m_{ji}$, such that $x' \in C^j$. But, in that case, by Property (N2), $\neg_j C \in \mathcal{L}_j(x')$, implying, by the induction hypothesis, that $x' \in (\neg_j C)^j = \Delta^j \setminus C^j$, a contradiction.
- For the last case involving negation, assume that $\neg_j C \in \mathcal{L}_i(x)$. Then by Property (N3), there exists $x' \in U_j$, with $(x', x) \in m_{ji}$ and $\neg_j C \in \mathcal{L}_j(x')$. Therefore, using the previous case, we get $x \in m_{ji}(x') \subseteq m_{ji}((\neg_j C)^j) = m_{ji}(\Delta^j \setminus C^j) = (\neg_j C)^i$.
- If $C_1 \sqcap_j C_2 \in \mathcal{L}_i(x)$, then, by Property (A1), there exists $x' \in U_j$, with $(x', x) \in m_{ji}$, such that $C_1 \in \mathcal{L}_j(x')$ and $C_2 \in \mathcal{L}_j(x')$. Therefore, using the induction hypothesis, $x \in m_{ji}(x') \subseteq m_{ji}(C_1^j \cap C_2^j) = m_{ji}((C_1 \sqcap_j C_2)^j) = (C_1 \sqcap_j C_2)^i$.
- The case $C = C_1 \sqcup_j C_2$ may be handled similarly, using Property (A2).
- If $\forall_j R.C \in \mathcal{L}_i(x)$, then, by Property (A3), there exists $x' \in U_j$, with $(x', x) \in m_{ji}$, such that, for all $y' \in U_j$, with $R \in \mathcal{L}_j(\langle x', y' \rangle)$, $C \in \mathcal{L}_j(y')$. Thus, by the definition of R^j , we get, using the induction hypothesis, that, there exists $x' \in U_j$, with $(x', x) \in m_{ji}$, such that $x' \in (\forall_j R.C)^j$. Hence $x \in m_{ji}((\forall_j R.C)^j) = (\forall_j R.C)^i$.
- Finally, suppose that $\exists_j R.C \in \mathcal{L}_i(x)$. Then, by Property (A4), there exist $x', y' \in U_j$, such that $(x', x) \in m_{ji}$, $R \in \mathcal{L}_j(\langle x', y' \rangle)$ and $C \in \mathcal{L}_j(y')$. Thus, again using the definition of R^j and the induction hypothesis, we get that, there exists $x' \in U_j$, with $(x', x) \in m_{ji}$, such that $x' \in (\exists_j R.C)^j$. This shows that $x \in m_{ji}(x') \subseteq m_{ji}((\exists_j R.C)^j) = (\exists_j R.C)^i$.

Notice, now, that $D^w \neq \emptyset$. In fact, by Property (D1), there exists $x \in U_w$, such that $D \in \mathcal{L}_w(x)$. Therefore, using Implication (2.2), $x \in D^w \neq \emptyset$. Finally, again using Implication (2.2), it is shown that, if $C \sqsubseteq D$ is an i -formula, then $C^i \subseteq D^i$. In fact, using Properties (D2) and (A1), we get that $\neg_i C \sqcup_i D \in \mathcal{L}_i(x)$. Thus, by Property (A2), either $\neg_i C \in \mathcal{L}_i(x)$ or $D \in \mathcal{L}_i(x)$. Therefore, by Implication (2.2), $x \notin C^i$ or $x \in D^i$, whence $C^i \subseteq D^i$, and, hence, $\mathcal{I}_i \models T_i$. ■

In Lemma 28, the converse is established, i.e., that, if an F- \mathcal{ALCT} concept D in a module T_w is satisfiable as witnessed by T_w , then it has a federated tableau with respect to T_w .

Lemma 28 *Let D be a concept in a module T_w of an F- \mathcal{ALCT} KB $T = \{T_i\}_{i \in V}$. If D is satisfiable as witnessed by T_w , then D has a federated tableau w.r.t. T_w .*

Proof:

Suppose that $\mathcal{I} = \langle \{\mathcal{I}_i\}_{i \in V_w}, \{r_{ij}\}_{(i,j) \in E_w} \rangle$ is a model of T_w^* , with $D^w \neq \emptyset$. A federated tableau $M = \langle \{M_i\}_{i \in V_w}, \{m_{ij}\}_{(i,j) \in E_w} \rangle$ for T_w^* , with $M_i = \langle U_i, F_i, \mathcal{L}_i \rangle$, may be defined as follows:

$$\begin{aligned}
 U_i &= \Delta^i; \\
 F_i &= \bigcup \{R^i : R \in \widehat{\mathcal{R}}_i\}; \\
 \mathcal{L}_w(x) &= \{C \in \text{sub}_w(D) \cup \text{sub}_w(C_{T_w}) : x \in C^w\}, \quad x \in \Delta^w; \\
 \mathcal{L}_w(\langle x, y \rangle) &= \{R \in \text{Rol}(D) \cup \text{Rol}(C_{T_w}) : \langle x, y \rangle \in R^w\}, \quad x, y \in \Delta^w; \\
 \mathcal{L}_i(x) &= \{C \in \text{sub}_i(C_{T_i}) : x \in C^i\}, \quad x \in \Delta^i, i \neq w; \\
 \mathcal{L}_i(\langle x, y \rangle) &= \{R \in \text{Rol}(C_{T_i}) : \langle x, y \rangle \in R^i\}, \quad x, y \in \Delta^i; \\
 m_{ij} &= r_{ij}.
 \end{aligned}$$

We now verify that M is indeed a tableau for D w.r.t. T_w , i.e., that it satisfies all conditions in the definition of a federated tableau (Conditions (D1)-(A4)).

(D1): Since $D^w \neq \emptyset$, there exists $x \in U_w$, such that $D \in \mathcal{L}_w(x)$.

(D2): Since \mathcal{I}_i is a model of T_i , we have, for every $x \in U_i$, $x \in C_{T_i}^i$, whence $C_{T_i} \in \mathcal{L}_i(x)$.

- (B1): Suppose $C \in \widehat{\mathcal{C}}_i \cap (\mathcal{C}_j \cup \{\top_j\})$, $(j, i) \in E$. Then we have $C \in \mathcal{L}_i(x)$ iff, by the definition of $\mathcal{L}_i(x)$, $x \in C^i = r_{ji}(C^j) = m_{ji}(C^j)$ iff, there exists $x' \in U_j$, with $(x', x) \in m_{ji}$, such that $x' \in C^j$, iff, there exists $x' \in U_j$, with $(x', x) \in m_{ji}$, such that $C \in \mathcal{L}_j(x')$.
- (B2): Suppose that $R \in \widehat{\mathcal{R}}_i \cap \mathcal{R}_j$. Then $R \in \mathcal{L}_i(\langle x, y \rangle)$ iff $(x, y) \in R^i = r_{ji}(R^j) = m_{ji}(R^j)$ if and only if, there exist $x', y' \in U_j$, with $(x', x), (y', y) \in m_{ji}$, such that $(x', y') \in R^j$ iff $R \in \mathcal{L}_j(\langle x', y' \rangle)$.
- (N1): If $C \in \mathcal{L}_i(x)$, such that $\neg_i C$ is of type 1, then $x \in C^i$, whence $x \notin \Delta^i \setminus C^i = (\neg_i C)^i$. Thus $\neg_i C \notin \mathcal{L}_i(x)$.
- (N2): Suppose $\neg_i C \in \mathcal{L}_i(x)$ is of type 2 with trace j and $x' \in U_j$, with $(x', x) \in m_{ji} = r_{ji}$. For the sake of obtaining a contradiction, suppose that $\neg_j C \notin \mathcal{L}_j(x')$. Then $x' \notin (\neg_j C)^j = \Delta^j \setminus C^j$, i.e., $x' \in C^j$. Therefore, $x \in r_{ji}(C^j)$, whence $x \notin \Delta^i \setminus r_{ji}(C^j) = (\neg_i C)^i$. This yields $\neg_i C \notin \mathcal{L}_i(x)$, which contradicts our hypothesis.
- (N3): Finally, suppose that $C \in \widehat{\mathcal{C}}_i \cap \widehat{\mathcal{C}}_j$, $(j, i) \in E$, $j \neq i$, with $\neg_j C \in \mathcal{L}_i(x)$. Thus $x \in (\neg_j C)^i = r_{ji}(\Delta^j \setminus C^j)$. Thus, there exists $x' \in U_j$, with $(x', x) \in r_{ji} = m_{ji}$, such that $x' \notin C^j$. But, then, $x' \in \Delta^j \setminus C^j = (\neg_j C)^j$, whence $\neg_j C \in \mathcal{L}_j(x')$.
- (A1): If $C_1 \sqcap_j C_2 \in \mathcal{L}_i(x)$, then $x \in (C_1 \sqcap_j C_2)^i = r_{ji}(C_1^j \cap C_2^j)$. Thus, there exists $x' \in U_j$, with $(x', x) \in r_{ji} = m_{ji}$, such that $x' \in C_1^j$ and $x' \in C_2^j$, i.e., such that $C_1 \in \mathcal{L}_j(x')$ and $C_2 \in \mathcal{L}_j(x')$.
- (A2): This case is handled very similarly to the previous one.
- (A3): Suppose that $\forall_j R.C \in \mathcal{L}_i(x)$. Then

$$x \in (\forall_j R.C)^i = r_{ji}(\{x' \in \Delta^j : (\forall y' \in \Delta^j)((x', y') \in R^j \rightarrow y' \in C^j)\}).$$

This means that there exists $x' \in \Delta^j = U_j$, with $(x', x) \in r_{ji} = m_{ji}$, such that, for all $y' \in \Delta^j = U_j$, with $(x', y') \in R^j$, i.e., $R \in \mathcal{L}_j(\langle x', y' \rangle)$, $y' \in C^j$, i.e., $C \in \mathcal{L}_j(y')$.

(A4): Finally, suppose that $\exists_j R.C \in \mathcal{L}_i(x)$. Then

$$x \in (\exists_j R.C)^i = r_{ji}(\{x' \in \Delta^j : (\exists y' \in \Delta^j)((x', y') \in R^j \text{ and } y' \in C^j)\}).$$

Thus, there exists $x' \in \Delta^j = U_j$, with $(x', x) \in r_{ji} = m_{ji}$, such that, there exists a $y' \in \Delta^j = U_j$, with $(x', y') \in R^j$, i.e., $R \in \mathcal{L}_j(\langle x', y' \rangle)$, and $y' \in C^j$, i.e., $C \in \mathcal{L}_j(y')$.

■

By combining Lemmas 27 and 28, we obtain the first main result of the paper establishing the equivalence between satisfiability and the existence of a tableau.

Theorem 29 *Let $T = \{T_i\}_{i \in V}$ be a F- \mathcal{ALCI} KB and D be a concept in module T_w . Then D is satisfiable as witnessed by T_w iff D has a federated tableau with respect to T_w .*

2.5 Tableau Algorithm for F- \mathcal{ALCI}

We now proceed to describe a sound and complete algorithm to determine the existence of a tableau for an F- \mathcal{ALCI} concept D with respect to a witness module T_w . The algorithm allows a local tableau to be created and maintained by a local reasoner. Thus, reasoning is carried out by a federation of reasoners that communicate with each other via messages instead of a single reasoner over an integrated ontology. Some implementation details, especially those concerning synchronization issues of the federated reasoners, are omitted.

2.5.1 Federated Completion Graph

The algorithm works on a dynamically evolving federated completion graph, which is a partial finite description of a tableau. A **federated completion graph** is a set $G = \{G_i\}_{i \in V_w}$, of local completion graphs. A **local completion graph** $G_i = \langle V_i, E_i, \mathcal{L}_i \rangle$, $i \in V_w$, consists of a finite set of finite trees, i.e., a forest, where V_i and E_i are the corresponding sets of nodes and edges respectively, and of a function \mathcal{L}_i , that assigns labels to nodes and edges in G_i , exactly as was the case with local tableaux. Each node x in V_i represents an individual in the

corresponding tableau, denoted $i : x$, and is labeled with $\mathcal{L}_i(x)$, a set of concepts of which x is a member. Each edge $\langle x, y \rangle \in E_i$ represents an edge in the tableau, and is labeled with $\mathcal{L}_i(\langle x, y \rangle)$, the set of roles of which it is an instance.

If $R \in \mathcal{L}_i(\langle x, y \rangle)$ or $R^- \in \mathcal{L}_i(\langle y, x \rangle)$, y is said to be a **local R -successor** of x and x is said to be a **local R -predecessor** of y . **Local ancestors** and **local descendants** of a node are defined in the usual manner.

Every node x has associated with it a set of nodes $\text{org}(x)$, which, informally speaking, are the nodes from which x is related via image domain relations. If $(i : x) \in \text{org}(j : y)$ and $(i, j) \in E_w$, we say that node $y \in V_j$ is an **image** of node $x \in V_i$, that node x is a **pre-image** of node y , and that there is a **graph relation** $\langle x, y \rangle$.

A typical federated completion graph consists of local successor relations in local forests together with graph relations across forests in different local reasoners. To construct a model for the ontology resulting by integrating all modules, as was done in [6], a different technique is used here. One keeps all forests disjoint, but uses the graph relations to map nodes in one forest to nodes in other forests.

2.5.2 Federated Tableau Expansion

A federated F- \mathcal{ALCI} completion graph is constructed by applying a set of tableau expansion rules and by exchanging messages between local reasoners. The F- \mathcal{ALCI} expansion rules are adapted from the \mathcal{ALCI} expansion rules. The label of each node in each local completion graph G_i will contain C_{T_i} , the internalization of T_i . A local completion graph can create images or pre-images of its local nodes in another local completion graph, as needed, during an expansion.

As in the tableau algorithm for \mathcal{ALCI} , some nodes in the graph may be *blocked*. The exact definition, whose main motivation is the detection of cycles in tableau expansions, is as follows:

Definition 30 (Equality Blocking) *For a federated completion graph of an F- \mathcal{ALCI} ontology, a node x is **directly blocked** by a node y , if both x and y are in the same local completion graph G_i , for some i , y is a local ancestor of x , and $\mathcal{L}_i(x) = \mathcal{L}_i(y)$. Node x is **indirectly***

blocked by a node y if one of x 's local ancestors is directly blocked by y . Node x is **blocked** by y if it is directly or indirectly blocked by y .

Equality blocking in F- \mathcal{ALCT} only depends on the *local* information in completion graphs, i.e., a node is blocked only by its *local* ancestors.

A **concept reporting message** creates image or pre-image nodes and/or propagates concept labels of a node to the corresponding image node or pre-image node. We use $S += X$ to denote the operation of adding the elements of the set X to a set S , i.e., the operation $S = S \cup X$. We have five kinds of concept reporting messages and each of these messages may be transmitted *only once*.

- A **forward concept reporting message** $r^{i \rightarrow j}(x, C)$ executes the following action: **if** there exists $x' \in V_j$, such that $x \in \text{org}(x')$ and $C \notin \mathcal{L}_j(x')$, **then** $\mathcal{L}_j(x') += \{C\}$.
- A **soft backward concept reporting message** $r^{j \leftarrow i}(x, C)$ executes the following actions: **if** $x' \in V_j$, with $x' \in \text{org}(x)$, **then** $\mathcal{L}_j(x') += \{C\}$, **if** $C \notin \mathcal{L}_j(x')$.
- A **backward concept reporting message** $r^{j \leftarrow i}(x, C)$ executes the following action: create an $x' \in V_j$, with $x' \in \text{org}(x)$ and $\mathcal{L}_j(x') = \{C\}$.
- A **forward role reporting message** $r^{i \rightarrow j}(\langle x, y \rangle, R)$ executes the following action: **if** there exist $x', y' \in V_j$, such that $x \in \text{org}(x')$, $y \in \text{org}(y')$ and $R \notin \mathcal{L}_j(\langle x', y' \rangle)$, **then** $\mathcal{L}_j(\langle x', y' \rangle) += \{R\}$.
- A **backward role reporting message** $r^{j \leftarrow i}(\langle x, y \rangle, R)$ executes the following action: create $x', y' \in V_j$, with $x' \in \text{org}(x)$, $y' \in \text{org}(y)$, and set $\mathcal{L}_j(\langle x', y' \rangle) = \{R\}$.

The expansion rules are:

- A rule ensuring that every element in G_i satisfies C_{T_i} .
 - **D-rule:** **if** $C_{T_i} \notin \mathcal{L}_i(x)$, **then** $\mathcal{L}_i(x) += \{C_{T_i}\}$.
- Four rules imposing forward and backward concept and role compatibilities:

- FCN-rule: **if** $C \in \mathcal{L}_j(x)$, $C \in \widehat{\mathcal{C}}_i \cap (\mathcal{C}_j \cup \{\top_j\})$, $(j, i) \in E$, and x is not blocked, **then** transmit $r^{j \rightarrow i}(x, C)$.
- BCN-rule: **if** $C \in \mathcal{L}_i(x)$, $C \in \widehat{\mathcal{C}}_i \cap (\mathcal{C}_j \cup \{\top_j\})$, $(j, i) \in E$, and x is not blocked, **then** transmit $r^{j \leftarrow i}(x, C)$.
- FRN-rule: **if** $R \in \mathcal{L}_j(\langle x, y \rangle)$, $R \in \widehat{\mathcal{R}}_i \cap \mathcal{R}_j$, $(j, i) \in E$, and x or y are not blocked, **then** transmit $r^{j \rightarrow i}(\langle x, y \rangle, R)$.
- BRN-rule: **if** $R \in \mathcal{L}_i(\langle x, y \rangle)$, $R \in \widehat{\mathcal{R}}_i \cap \mathcal{R}_j$, $(j, i) \in E$, and x or y are not blocked, **then** transmit $r^{j \leftarrow i}(\langle x, y \rangle, R)$.
- Two negation rules (a local and a foreign one):
 - $L\neg$ -rule: **if** $\neg_i C \in \mathcal{L}_i(x)$ is of type 2 with trace j , $C \in \widehat{\mathcal{C}}_i \cap \widehat{\mathcal{C}}_j$, $(j, i) \in E$, $i \neq j$, and x is not blocked, **then** transmit $r^{j \leftarrow i}(x, \text{nlf}_j(\neg_j C))$.
 - $F\neg$ -rule: **if** $\neg_j C \in \mathcal{L}_i(x)$, $C \in \widehat{\mathcal{C}}_i \cap \widehat{\mathcal{C}}_j$, $(j, i) \in E$, $i \neq j$, and x is not blocked, **then** transmit $r^{j \leftarrow i}(x, \text{nlf}_j(\neg_j C))$.
- Two conjunction rules (a local and a foreign one):
 - $L\sqcap$ -rule: **if** $C_1 \sqcap_i C_2 \in \mathcal{L}_i(x)$, x is not blocked, and $\{C_1, C_2\} \not\subseteq \mathcal{L}_i(x)$, **then** $\mathcal{L}_i(x) += \{C_1, C_2\}$.
 - $F\sqcap$ -rule: **if** $C_1 \sqcap_j C_2 \in \mathcal{L}_i(x)$, $(j, i) \in E$, $j \neq i$, and x is not blocked, **then** transmit $r^{j \leftarrow i}(x, C_1 \sqcap_j C_2)$.
- Two disjunction rules (a local and a foreign one):
 - $L\sqcup$ -rule: **if** $C_1 \sqcup_i C_2 \in \mathcal{L}_i(x)$, x is not blocked, and $\{C_1, C_2\} \cap \mathcal{L}_i(x) = \emptyset$, **then** $\mathcal{L}_i(x) += \{C_1\}$ or $\mathcal{L}_i(x) += \{C_2\}$.
 - $F\sqcup$ -rule: **if** $C_1 \sqcup_j C_2 \in \mathcal{L}_i(x)$, $(j, i) \in E$, $j \neq i$, and x is not blocked, **then** transmit $r^{j \leftarrow i}(x, C_1 \sqcup_j C_2)$.
- Two universal quantification rules (a local and a foreign one):

- L \forall -rule: **if** $\forall_i R.C \in \mathcal{L}_i(x)$, x is not blocked and, there exists $y \in V_i$, with $R \in \mathcal{L}_i(\langle x, y \rangle)$, **then** $\mathcal{L}_i(y) += \{C\}$, if $C \notin \mathcal{L}_i(y)$.
- F \forall -rule: **if** $\forall_j R.C \in \mathcal{L}_i(x)$, $(j, i) \in E$, $j \neq i$, and x is not blocked, **then** transmit $r^{j \leftarrow i}(x, \forall_j R.C)$.
- Two existential quantification rules (a local and a foreign one):
 - L \exists -rule: **if** $\exists_i R.C \in \mathcal{L}_i(x)$ and x is not blocked and x has no local R -successor y , with $C \in \mathcal{L}_i(y)$, **then** create a new node $y \in V_i$ and set $\mathcal{L}_i(\langle x, y \rangle) = \{R\}$ and $\mathcal{L}_i(y) = \{C\}$.
 - F \exists -rule: **if** $\exists_j R.C \in \mathcal{L}_i(x)$, $(j, i) \in E$, $j \neq i$, and x is not blocked, **then** transmit $r^{j \leftarrow i}(x, \exists_j R.C)$.

All the rules presented above correspond to properties that the federated tableau must satisfy. The D-rule makes sure that Property (D2) is satisfied. The FCN (Forward Concept Name) and BCN (Backward Concept Name) rules ensure that Property (B1) of a tableau is satisfied by the completion graphs. Similarly, the FRN (Forward Role Name) and BRN (Backward Role Name) rules take care of Property (B2). The negation, conjunction, disjunction, universal and existential quantification rules have both an L (Local) and an F (Foreign) version. These ten rules collectively make sure that all properties pertaining to negation, conjunction, disjunction and the quantifiers, i.e., Properties (N1)-(A4), of a tableau are satisfied by the completion graphs.

A federated completion graph is **complete** if no F- \mathcal{ALCI} expansion rule can be applied to it, and it is **clash-free** if there is no x in any local completion graph G_i , such that both C and $\neg_i C$ are in $\mathcal{L}_i(x)$, for some concept C .

For a satisfiability query of a concept D as witnessed by a module T_w , a local completion graph G_w , with an initial node x_0 , with $\mathcal{L}_w(x_0) = \{D\}$, will be created first. The F- \mathcal{ALCI} tableau expansion rules will be applied until a complete and clash-free federated completion graph is found or until all search efforts for such a federated completion graph fail. Then, simi-

larly to ordinary DL tableau, D will be satisfiable from the point of view of T_w if the expansion rules yield a complete, clash-free federated completion graph, and unsatisfiable otherwise.

Example 4: We present a very simple example to illustrate how some of the expansion rules and some of the concept reporting messages work in the algorithm. Suppose that the underlying graph G is the complete graph on two nodes, 1 and 2. (For the purposes of this example, we overlook the restriction that G be acyclic.) Module T_1 , corresponding to node 1, has two concept names A, B and module T_2 , corresponding to node 2, does not have any concept names. Suppose that T_1 consists of the single subsumption $A \sqsubseteq B$ and that we want to check the satisfiability of $\neg_2 A \sqcup_1 B$ from the point of view of the second module. We define $C_{T_1} = \top_1 \sqcap_1 (\neg_1 A \sqcup_1 B)$ and the algorithm is initialized by creating a node x_0 in G_2 , such that $\mathcal{L}_2(x_0) = \{\neg_2 A \sqcup_1 B\}$. (See Figure 2.1.)

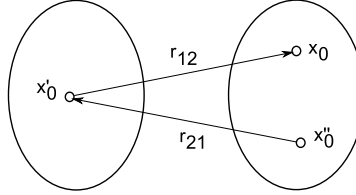


Figure 2.1 The distributed model of Example 4.

The $F\sqcup$ -rule applies, whence a concept reporting message $r^{1\leftarrow 2}(x_0, \neg_2 A \sqcup_1 B)$ is sent from module 2 to module 1. This creates a node x'_0 in G_1 , with $\mathcal{L}_1(x'_0) = \{\neg_2 A \sqcup_1 B\}$. We also have that $\text{org}(x_0) = \{x'_0\}$.

Next, the D -rule is applied to x'_0 , whence $\mathcal{L}_1(x'_0) = \{\top_1 \sqcap_1 (\neg_1 A \sqcup_1 B), \neg_2 A \sqcup_1 B\}$, and, then, the $L\sqcap$ -rule and twice the $L\sqcup$ -rule apply to obtain

$$\mathcal{L}_1(x'_0) = \{\top_1 \sqcap_1 (\neg_1 A \sqcup_1 B), \neg_2 A \sqcup_1 B, \top_1, \neg_1 A \sqcup_1 B, \neg_1 A, \neg_2 A\}.$$

Finally, the occurrence of $\neg_2 A$ triggers an application of the $F\neg$ -rule, which causes the transmission of a $r^{2\leftarrow 1}(x'_0, \neg_2 A)$ concept reporting message. A new node x''_0 is created in G_2 , with label $\mathcal{L}_2(x''_0) = \{\neg_2 A\}$ and $\text{org}(x'_0) = \{x''_0\}$. As a consequence of this the $L\neg$ -rule is applied in G_2 and an $r^{1\leftarrow -2}(x_0, \neg_1 A)$ concept reporting message is delivered. This message, however,

does not cause any changes because there is no x in G_1 , such that $x \in \text{org}(x_0'')$.

The final federated model $\mathcal{I} = \langle \{\mathcal{I}_i\}_{i=1,2}, \{r_{ij}\}_{i,j=1,2} \rangle$, therefore, consists of $\Delta^1 = \{x_0'\}$, $\Delta^2 = \{x_0, x_0''\}$, with the interpretation $A^1 = B^1 = \emptyset$ and, for the image domain relations we get

$$r_{12} = \{(x_0', x_0)\}, \quad r_{21} = \{(x_0'', x_0')\}.$$

Since no clash occurred, the algorithm returns that $\neg_2 A \sqcup_1 B$ is satisfiable from the point of view of T_2 . Indeed we have:

$$\begin{aligned} (\neg_2 A \sqcup_1 B)^2 &= r_{12}((\neg_2 A \sqcup_1 B)^1) \\ &= r_{12}((\neg_2 A)^1 \cup B^1) \\ &= r_{12}(r_{21}(\Delta^2 \setminus A^2) \cup B^1) \\ &= r_{12}(r_{21}(\Delta^2 \setminus r_{12}(A^1)) \cup B^1) \\ &= r_{12}(r_{21}(\Delta^2) \cup B^1) \\ &= r_{12}(\{x_0'\}) \\ &= \{x_0\}. \end{aligned}$$

2.5.3 Synchronization

The federated tableau algorithm depends crucially on being able to synchronize the processes occurring in each of the local reasoners. It must be ensured that, at all times during the execution of the parallel threads of the algorithm, all local processes, i.e., all applications of local expansion rules in the various local completion graphs, refer to the same sequence of non-deterministic choices. This may be achieved in a variety of different ways. One method was presented in detail in [6] in the context of P-DLs. It uses clocks, timestamps and a token to achieve synchronization and to implement efficient backtracking when a clash occurs and another non-deterministic option has to be tried.

In the present paper, we prefer to give an abstract view of the synchronization process without providing a detailed description of either the synchronizing entities or the specific steps. There are many possible choices and which one is adopted is a decision that can be

relegated to the implementation of the algorithm.

The guiding principle in building the local completion graphs in our case will also be the sequence of applications of the $L\sqcup$ -rules. Only one module is allowed to apply the $L\sqcup$ -rule at any particular time during the execution of the algorithm. Otherwise, various other rules may be applied by the participating modules simultaneously. A newly generated concept label or role label is accompanied by a tag indicating the latest application of the $L\sqcup$ -rule before its creation. New labels can be created not only by the local rules, but also by the foreign rules that involve concept and role reporting message transmissions. If a clash is detected in a local completion graph, then all labels that have been created after the last non-deterministic choice will be deleted and all nodes or edges without labels will be purged. This returns the algorithm to the same state that preceded the latest application of an $L\sqcup$ -rule. Another non-deterministic choice that has not been applied before will be made and the process will start again. This “pruning operation” is necessary to restore all local completion graphs to their status just before the choice which led to the clash, or to the initial status of the local tableau, if no choice at all was ever made.

Note that local completion graphs may perform expansions on different reasoning subtasks concurrently. This improves the overall efficiency and scalability of the reasoning process. Further, note that with the introduction of messages, equality blocking in $F\text{-}\mathcal{ALCI}$ is *dynamic*: it can be established, broken and re-established. Moreover, the completeness of a local completion graph is also dynamic. A complete local completion graph may become incomplete, i.e., some expansion rules may become applicable, when a new reporting message arrives.

2.6 Correctness and Complexity

In order to show that the algorithm is a decision procedure for concept satisfiability in $F\text{-}\mathcal{ALCI}$, it is necessary to prove that the algorithm terminates, that the models that can be constructed from clash-free and complete federated completion graphs, generated from the algorithm, are valid with respect to the semantics of the logic (soundness) and that the algorithm always finds a model if one exists (completeness).

Termination and complexity of the algorithm is obtained by proving that there is an upper bound for the total size of all local completion graphs. We use the following notation throughout the analysis of the algorithm:

$$\begin{aligned} n_i &= |C_{T_i}|, \quad i \neq w; & n_w &= |C_{T_w}| + |D|; \\ s_i &= \sum_{(i,j) \in E} n_j, \quad i \neq w; & s_w &= n_w, \end{aligned}$$

where n_i is the combined length of all i -formulas in T_i in negation local form and s_i is the sum of the n_j 's for all j 's, such that T_j imports T_i . The reason why s_i is important in the analysis of the federated algorithm is that all j -axioms that contain an i -connective will cause a foreign rule to send a backward concept reporting or a backward role reporting message to P_i to be processed. More specifically, we have the following lemma:

Lemma 31 *Let $\Sigma = \{T_i\}_{i \in V}$ be an $F\text{-}\mathcal{ALCI}$ KB, $D \in \widehat{\mathcal{C}}_w$ and $m = \sum_{i \in V_w} n_i$. The $F\text{-}\mathcal{ALCI}$ tableau algorithm runs in worst case non-deterministic $O(2^m \cdot \prod_{j \in V_w} 2^{2^{s_j} \log s_j})$ time.*

Proof:

We define $f(x) = 2^{2^x \log x}$. We start with a set of observations:

- *For every node that has no local predecessor (called **local top node** henceforth), its local descendants have a tree shape.* This observation follows from the form of the expansion rules.
- *For every local top node $j : x$, $j \neq w$, x must be a preimage of a node in another local completion graph G_i , such that $(j, i) \in E_w$.* This holds because such an x must be created by either a backward concept reporting message triggered by an application of the BCN-rule or of a foreign rule or by a backward role reporting message triggered by an application of the BRN-rule.
- *For all j, x , all local descendants of $j : x$ in G_j , that are not successors, are not preimages of nodes in any other local completion graph.* This holds because a local descendant of $j : x$, that is not a successor, is generated only by an application of the $L\exists$ -rule, while a

preimage node is created only by an application of the the BCN, the BRN or a foreign rule.

Hence, 1) each local completion graph is a forest; 2) the root of every tree, i.e., a local top node in a local completion graph, except for the root of G_w , is a preimage of a node in another local completion graph.

Next we prove that the size of each local completion graph, hence also the total size of the “global completion graph”, is limited.

First, due to equality blocking, for any local top node in G_j , the depth of its local descendant tree is bounded by $O(2^{s_j})$ and its breadth is bounded by the number of “ \exists_j ” in all $\bigcup_{(j,i) \in E} C_{T_i}$, for $j \neq w$, or in $C_{T_w} \sqcup D$, for $j = w$, which is smaller than s_j . Thus, the size of the tree is bounded by $O(s_j 2^{s_j}) < O(f(s_j))$.

Since there is only acyclic importing, we can put all modules $\{T_i\}_{i \in V_w}$ in an ordered list \mathfrak{L} , such that $\mathfrak{L}_1 = T_w$ and each module T_i comes in \mathfrak{L} before all modules $\{T_j\}_{j \in V_i}$, in a way similar to topological sorting in DAG. Let $\#(\mathfrak{L}_j)$ be the subscript of the module at \mathfrak{L}_j . Then, we have that the size of $G_{\#(\mathfrak{L}_j)}$ is bounded by:

$$\begin{aligned} |G_{\#(\mathfrak{L}_1)}| &: O(f(s_w)) \\ |G_{\#(\mathfrak{L}_j)}| &: O\left(\sum_{k < j} |G_{\#(\mathfrak{L}_k)}| \times f(s_{\#(\mathfrak{L}_j)})\right), \text{ for } j > 1 \end{aligned}$$

This holds because there is only one local top node in $G_{\#(\mathfrak{L}_1)} = G_w$ (the original node), and, for every $j > 1$ and $p = \#(\mathfrak{L}_j)$, the number of local top nodes in G_p is limited by $\sum_{(p,q) \in E_w} |G_q|$, i.e., by the total size of the local completion graphs of modules that directly import T_p , since all nodes in T_p must be preimage nodes of nodes in those local completion graphs. In the worst case, $\{T_q : (p, q) \in E_w\}$ contains all modules that are before j in \mathfrak{L} . On the other hand, the size of a tree under a local top node in G_k is limited by $f(s_k)$.

Setting $|G_{\#(\mathfrak{L}_j)}| = t_j$ and $e_j = f(s_{\#(\mathfrak{L}_j)})$, we obtain that t_j is bounded by

$$O((t_1 + t_2 + \dots + t_{j-1}) \times e_j). \quad (2.3)$$

Using induction, it will now be shown that t_j is bounded by

$$O(2^{j-2} \times e_1 \times \dots \times e_j), \text{ for } j > 1. \quad (2.4)$$

By Equation (2.3), when $j = 2$, t_2 is bounded by $O(t_1 \times e_2) = O(e_1 \times e_2)$, whence Equation (2.4) holds. Let $j > 2$. Assuming, as the induction hypothesis, that, for every $1 < k < j$, Equation (2.4) holds, we have, by Equation (2.3), that t_j is bounded by

$$\begin{aligned} O((t_1 + t_2 + \dots + t_{j-1}) \times e_j) &< O((e_1 + 2^0 e_1 e_2 + \dots + 2^{j-3} e_1 e_2 \dots e_{j-1}) e_j) \\ &< O((1 + 2^0 + \dots + 2^{j-3}) \times e_1 e_2 \dots e_j) \\ &= O(2^{j-2} e_1 e_2 \dots e_j) \end{aligned}$$

This finishes the induction step and concludes the proof of Equation (2.4). Hence, the size of all local completion graphs is bounded by:

$$\begin{aligned} O\left(e_1 + \sum_{2 \leq j \leq m} \left(2^{j-2} \prod_{k \leq j} e_k\right)\right) &\leq O\left(2^{m-1} \times \prod_{j \in V_w} f(s_j)\right) \\ &< O\left(2^m \times \prod_{j \in V_w} 2^{s_j \times \log s_j}\right) \end{aligned}$$

■

Lemma 31 leads to the following theorem on the complexity of the federated algorithm for deciding F- \mathcal{ALCI} concept satisfiability.

Theorem 32 (Termination and Complexity) *Let Σ be an F- \mathcal{ALCI} ontology and $D \in \widehat{\mathcal{C}}_w$. The F- \mathcal{ALCI} tableau algorithm runs in worst case 2NEXPTIME w.r.t. the size of D and the sum of the sizes of the modules in $\{T_i\}_{i \in V_w}$.*

Proof:

Let $s = \max\{s_i : i \in V_w\}$. In general, $m \ll 2^{s \log s}$. By Lemma 31, it follows that the total

size of all local completion graphs is bounded by

$$O\left(2^m \cdot 2^{m2^{(s+|D|)\log(s+|D|)}}\right) < O\left(2^{2^{(s+|D|)^2}}\right).$$

■

In the following two lemmas, soundness and completeness of the F- \mathcal{ALCI} algorithm are stated.

Theorem 33 (Soundness) *If the F- \mathcal{ALCI} algorithm yields a complete and clash-free federated completion graph for a concept D w.r.t. a witness module T_w , then D has a federated tableau w.r.t. T_w .*

Proof:

Let $G = \{G_i\}$, with $G_i = (V_i, E_i, \mathcal{L}_i^g)$, be a complete and clash-free federated completion graph generated by the F- \mathcal{ALCI} algorithm. We will obtain a tableau by “unraveling” blocked nodes and tableau relations. For a directly blocked node x , we denote by $\text{bk}(x)$ the node that directly blocks x . Thus, we have $\mathcal{L}_i^g(x) = \mathcal{L}_i^g(\text{bk}(x))$. We define a tableau $M = \langle \{M_i\}_{i \in V_w}, \{m_{ij}\}_{(i,j) \in E_w} \rangle$, with $M_i = \langle U_i, F_i, \mathcal{L}_i^m \rangle$, for D w.r.t. T_w in the following way:

$$\begin{aligned} U_i &= \{x \in V_i : x \text{ is not blocked}\}; \\ F_i &= E_i \upharpoonright_{U_i^2}; \\ \mathcal{L}_i^m(x) &= \mathcal{L}_i^g(x); \\ \mathcal{L}_i^m(\langle x, y \rangle) &= \mathcal{L}_i^g(\langle x, y \rangle) \cup \bigcup_{z: y = \text{bk}(z)} \mathcal{L}_i^g(\langle x, z \rangle); \\ m_{ij} &= \{\langle x, y \rangle \in U_i \times U_j \mid x \in \text{org}(y)\}, \text{ for } (i, j) \in E_w. \end{aligned}$$

We show that M satisfies all tableau properties.

(D1): Since $x_0 \in V_w$, x_0 is not blocked (it does not have any ancestors), and $D \in \mathcal{L}_w^g(x_0)$, we get that $x_0 \in U_w$ and $D \in \mathcal{L}_w^m(x_0)$.

(D2): Property (D2) holds because of the D-rule.

(B1): Suppose, first, that there exists $x' \in U_j$, with $(x', x) \in m_{ji}$, such that $C \in \mathcal{L}_j^m(x')$.

Then $x' \in U_j$ is not blocked and $x' \in \text{org}(x)$, whence, since $C \in \mathcal{L}_j^g(x')$, we get, by the FCN-rule, $C \in \mathcal{L}_i^g(x)$, i.e., $C \in \mathcal{L}_i^m(x)$ and the “if” direction of Property (B1) holds.

Suppose, conversely, that $C \in \mathcal{L}_i^m(x)$. Then $C \in \mathcal{L}_i^g(x)$ and x is not blocked, whence, by the BCN-rule, there exists $x' \in V_j$, which is not blocked because it is a local top node, such that $x' \in \text{org}(x)$ and $C \in \mathcal{L}_j^m(x')$. Therefore, $(x', x) \in m_{ji}$ and $C \in \mathcal{L}_j^g(x')$. Therefore, Property (B1) holds.

(B2): Suppose, first, that there exists $x', y' \in U_j$, with $(x', x), (y', y) \in m_{ji}$, such that $R \in \mathcal{L}_j^m(\langle x', y' \rangle)$. Then $x', y' \in U_j$ are not blocked and $x' \in \text{org}(x), y' \in \text{org}(y)$ and $R \in \mathcal{L}_j^g(\langle x', y' \rangle)$. Thus, by the FRN-rule, $R \in \mathcal{L}_i^g(\langle x, y \rangle)$, i.e., $R \in \mathcal{L}_i^m(\langle x, y \rangle)$ and the “if” direction of Property (B2) holds.

Suppose, conversely, that $R \in \mathcal{L}_i^m(\langle x, y \rangle)$. Then x is not blocked and $R \in \mathcal{L}_i^g(\langle x, y \rangle)$. Thus, by the BRN-rule, there exists $x', y' \in V_j$, with $x' \in \text{org}(x)$ and $y' \in \text{org}(y)$, such that $R \in \mathcal{L}_j^g(\langle x', y' \rangle)$ and x', y' cannot be blocked. Hence $R \in \mathcal{L}_j^m(\langle x', y' \rangle)$ and Property (B2) holds.

(N1): This property follows directly by the hypothesis that G is a clash-free federated completion graph.

(N2): Suppose $\neg_i C \in \mathcal{L}_i^m(x)$ is of type 2 with trace j and that $x' \in U_j$, with $(x', x) \in m_{ji}$. Then $\neg_i C \in \mathcal{L}_i^g(x)$, $x' \in V_j$, with $x' \in \text{org}(x)$ and neither x nor x' are blocked. Hence, by the $L\neg$ -rule, $\neg_j C \in \mathcal{L}_j^g(x')$. Therefore, $\neg_j C \in \mathcal{L}_j^m(x')$ and Property (N2) holds.

(N3): Suppose that $\neg_j C \in \mathcal{L}_i^m(x)$. Then $\neg_j C \in \mathcal{L}_i^g(x)$. Thus, by the $F\neg$ -rule, there exists $x' \in V_j$, with $x' \in \text{org}(x)$, such that $\neg_j C \in \mathcal{L}_j^g(x')$. This shows that, there exists $x' \in U_j$, such that $(x', x) \in m_{ji}$ and $\neg_j C \in \mathcal{L}_j^m(x')$. So Property (N3) holds.

(A1): Suppose that $C_1 \sqcap_j C_2 \in \mathcal{L}_i^m(x)$. Then $C_1 \sqcap_j C_2 \in \mathcal{L}_i^g(x)$. Therefore, if $j = i$, by the $L\sqcap$ -rule, we get that $C_1, C_2 \in \mathcal{L}_i^g(x)$, whence $C_1, C_2 \in \mathcal{L}_i^m(x)$. On the other hand, if $j \neq i$, then, by the $F\sqcap$ -rule, there exists $x' \in V_j$, with $x' \in \text{org}(x)$, such that $C_1 \sqcap_j C_2 \in \mathcal{L}_j^g(x')$.

Hence, by the previous case, we get that $C_1, C_2 \in \mathcal{L}_j^g(x')$, showing that, there exists $x' \in U_j$, with $(x', x) \in m_{ji}$, such that $C_1, C_2 \in \mathcal{L}_j^m(x')$. Thus, Property (A1) holds.

(A2): The proof of this case is very similar to that of Property (A1).

(A3): Suppose that $\forall_j R.C \in \mathcal{L}_i^m(x)$. Then $\forall_j R.C \in \mathcal{L}_i^g(x)$. If $j = i$ and $R \in \mathcal{L}_i^m(\langle x, y \rangle)$, then we have that $R \in \mathcal{L}_i^g(\langle x, y \rangle)$, whence, by the $L\forall$ -rule, $C \in \mathcal{L}_i^g(y)$, showing that $C \in \mathcal{L}_i^m(y)$. If, on the other hand, $j \neq i$, we get, by the $F\forall$ -rule, that there exists $x' \in V_j$, with $x' \in \text{org}(x)$, with $\forall_j R.C \in \mathcal{L}_j^g(x')$. Thus, by the $L\forall$ -rule, as applied in the previous case, for all $y' \in U_j$, with $R \in \mathcal{L}_j^m(\langle x', y' \rangle)$, we get that $C \in \mathcal{L}_j^m(y')$.

(A4): Suppose that $\exists_j R.C \in \mathcal{L}_i^m(x)$. Then $\exists_j R.C \in \mathcal{L}_i^g(x)$. If $j = i$, then, by the $L\exists$ -rule, there exists $y \in V_i$, such that $R \in \mathcal{L}_i^g(\langle x, y \rangle)$, with $C \in \mathcal{L}_i^g(y)$. Thus, in this case, $R \in \mathcal{L}_i^m(\langle x, y \rangle)$ and $C \in \mathcal{L}_i^m(y)$. If, on the other hand, $j \neq i$, we get, by the $F\forall$ -rule, that there exists $x' \in V_j$, with $x' \in \text{org}(x)$, with $\exists_j R.C \in \mathcal{L}_j^g(x')$. Thus, by the $L\exists$ -rule, as applied in the previous case, there exists $y' \in U_j$, with $R \in \mathcal{L}_j^g(\langle x', y' \rangle)$ and $C \in \mathcal{L}_j^g(y')$. Hence, we get that $R \in \mathcal{L}_j^m(\langle x', y' \rangle)$ and $C \in \mathcal{L}_j^m(y')$.

■

The following lemma shows that the federated algorithm is complete, i.e., that it always finds a complete and clash-free federated completion graph whenever there exists a federated tableau.

Theorem 34 (Completeness) *If a concept D has a federated tableau w.r.t. a witness module T_w of an $F\text{-}\mathcal{ALCI}$ KB $T = \{T_i\}_{i \in V}$, then the $F\text{-}\mathcal{ALCI}$ algorithm produces a complete and clash-free federated completion graph for D w.r.t. T_w .*

Proof:

Let $M = \langle \{M_i\}_{i \in V_w}, \{m_{ij}\}_{(i,j) \in E_w} \rangle$, with $M_i = \langle U_i, F_i, \mathcal{L}_i^m \rangle$, be a tableau for D w.r.t. T_w . We will use M to guide the application of the non-deterministic $L\sqcup$ -rule in a way that yields a complete and clash-free federated completion graph $G = \{G_i\}_{i \in V_w}$, with $G_i = (V_i, E_i, \mathcal{L}_i^g)$.

To construct G , we start with a single node x_0 in the local tableau M_w , with $D \in \mathcal{L}_w^m(x_0)$. Such an x_0 exists, since M is a tableau for D w.r.t. T_w . Let $\pi \subseteq \bigcup_{i \in V_w} (V_i \times U_i)$ be a function that maps all individuals in local completion graphs to individuals in corresponding local tableaux. Initially, we have $V_w = \{x_0\}$, $\mathcal{L}_w^g(x_0) = \{D\}$, $\pi(x_0) = x_0$ and all G_i , $i \neq w$, being empty. Next, we apply F- \mathcal{ALCI} expansion rules to extend G and π , in such a way that the following conditions always (inductively) hold:

$$\left\{ \begin{array}{l} \mathcal{L}_i^g(x) \subseteq \mathcal{L}_i^m(\pi(x)) \\ \text{if } R \in \mathcal{L}_i^g(\langle x, y \rangle), \text{ then } R \in \mathcal{L}_i^m(\langle \pi(x), \pi(y) \rangle) \\ \text{if } x \in \text{org}(y) \text{ in } G, \text{ then } \langle \pi(x), \pi(y) \rangle \in m_{ij}, \text{ for } (i, j) \in E_w \end{array} \right. \quad (2.5)$$

- D-rule: **if** $C_{T_i} \notin \mathcal{L}_i^g(x)$, **then** $\mathcal{L}_i^g(x) += \{C_{T_i}\}$. Since, by Property (D2), $C_{T_i} \in \mathcal{L}_i^m(\pi(x))$, this rule can be applied without violating Conditions (2.5).
- FCN-rule: **if** $C \in \mathcal{L}_j^g(x)$, x is not blocked, **then** transmit $r^{j \rightarrow i}(x, C)$, i.e., if there exists $x' \in V_i$, such that $x \in \text{org}(x')$, then $C \in \mathcal{L}_i^g(x')$. In that case, by the induction hypothesis, $C \in \mathcal{L}_j^m(\pi(x))$ and $(\pi(x), \pi(x')) \in m_{ji}$, whence by Property (B1), we obtain that $C \in \mathcal{L}_i^m(\pi(x'))$. Thus, Conditions (2.5) are not violated.
- BCN-rule: **if** $C \in \mathcal{L}_i^g(x)$, **then** transmit $r^{j \leftarrow i}(x, C)$. This will create an $x' \in V_j$, with $x' \in \text{org}(x)$ and $C \in \mathcal{L}_j^g(x')$. Since $C \in \mathcal{L}_i^g(x)$, we get that $C \in \mathcal{L}_i^m(\pi(x))$, whence, by Property (B1) of a federated tableau, there exists $z \in U_j$, with $(\pi(x), z) \in m_{ji}$, with $C \in \mathcal{L}_j^m(z)$. Set $\pi(x') = z$. Then we have that $\mathcal{L}_j^g(x') = \{C\} \subseteq \mathcal{L}_j^m(z) = \mathcal{L}_j^m(\pi(z))$. Moreover, we get $(\pi(x), \pi(x')) = (\pi(x), z) \in m_{ji}$ and, therefore, Conditions (2.5) are not violated.
- FRN-rule: **if** $R \in \mathcal{L}_j^g(\langle x, y \rangle)$, x or y not blocked, **then** transmit $r^{j \rightarrow i}(\langle x, y \rangle, R)$, i.e., if there exist x', y' , such that $x \in \text{org}(x'), y \in \text{org}(y')$, then $R \in \mathcal{L}_j^g(\langle x', y' \rangle)$. If $R \in \mathcal{L}_j^g(\langle x, y \rangle)$, then $R \in \mathcal{L}_i^m(\langle \pi(x), \pi(y) \rangle)$ and, if $x \in \text{org}(x'), y \in \text{org}(y')$, then $(\pi(x), \pi(x')), (\pi(y), \pi(y')) \in m_{ij}$, whence, by the tableau Property (B2), we must have $R \in \mathcal{L}_j^m(\langle \pi(x'), \pi(y') \rangle)$, whence Property (2.5) is not violated.

- **BRN-rule:** **if** $R \in \mathcal{L}_i^g(\langle x, y \rangle)$ and x or y are not blocked, **then** transmit $r^{j \leftarrow i}(\langle x, y \rangle, R)$, i.e., create $x', y' \in V_j$, with $x' \in \text{org}(x), y' \in \text{org}(y)$, such that $R \in \mathcal{L}_j^g(\langle x', y' \rangle)$. Since $R \in \mathcal{L}_i^g(\langle x, y \rangle)$, we get that $R \in \mathcal{L}_i^m(\langle \pi(x), \pi(y) \rangle)$. Therefore, by Property (B2), there exists $z, w \in U_j$, with $(z, \pi(x)) \in m_{ji}$ and $(w, \pi(y)) \in m_{ji}$, such that $R \in \mathcal{L}_j^m(\langle z, w \rangle)$. Set $\pi(x') = z$ and $\pi(y') = w$. Then, we get that $(\pi(x'), \pi(x)), (\pi(y'), \pi(y)) \in m_{ji}$ and $R \in \mathcal{L}_j^m(\langle \pi(x'), \pi(y') \rangle)$. Thus, Conditions (2.5) are not violated.
- **L \neg -rule:** **if** $\neg_i C \in \mathcal{L}_i^g(x)$ is of type 2 with trace j and x is not blocked, **then** transmit $r^{j \leftarrow i}(x, \neg_j C)$, i.e., if there exists $x' \in V_j$, with $x' \in \text{org}(x)$, then $\neg_j C \in \mathcal{L}_j^g(x')$. Under these circumstances, we have, by the induction hypothesis, that $\neg_i C \in \mathcal{L}_i^m(\pi(x))$ and $(\pi(x'), \pi(x)) \in m_{ji}$. Thus, by Property (N2), we get that $\neg_j C \in \mathcal{L}_j^m(\pi(x'))$, showing that Conditions (2.5) are not violated.
- **F \neg -rule:** **if** $\neg_j C \in \mathcal{L}_i^g(x)$ and x is not blocked, **then** transmit $r^{j \leftarrow i}(x, \neg_j C)$, i.e., create $x' \in V_j$, with $x' \in \text{org}(x)$, such that $\neg_j C \in \mathcal{L}_j^g(x')$. By the induction hypothesis, we have that $\neg_j C \in \mathcal{L}_i^m(\pi(x))$. Thus, by Property (N3), we get that, there exists $z \in U_j$, with $(z, \pi(x)) \in m_{ji}$, such that $\neg_j C \in \mathcal{L}_j^m(z)$. If we set $\pi(x') = z$, we get that $(\pi(x'), \pi(x)) = (z, \pi(x)) \in m_{ji}$ and $\neg_j C \in \mathcal{L}_j^m(\pi(x'))$. Hence, Conditions (2.5) are not violated.
- **L \sqcap -rule:** **if** $C_1 \sqcap_i C_2 \in \mathcal{L}_i^g(x)$ and x is not blocked, **then** $C_1, C_2 \in \mathcal{L}_i^g(x)$. In this case, by the induction hypothesis, $C_1 \sqcap_i C_2 \in \mathcal{L}_i^m(\pi(x))$. Thus, by Property (A1), we get that $C_1, C_2 \in \mathcal{L}_i^m(\pi(x))$, which shows that Conditions (2.5) are not violated.
- **F \sqcap -rule:** **if** $C_1 \sqcap_j C_2 \in \mathcal{L}_i^g(x)$ and x is not blocked, **then** transmit $r^{j \leftarrow i}(x, C_1 \sqcap_j C_2)$, i.e., create $x' \in V_j$, with $x' \in \text{org}(x)$, such that $C_1 \sqcap_j C_2 \in \mathcal{L}_j^g(x')$. In this case, by the induction hypothesis, $C_1 \sqcap_j C_2 \in \mathcal{L}_i^m(\pi(x))$. Thus, by Property (A1), there exists $z \in U_j$, with $(z, \pi(x)) \in m_{ji}$, such that $C_1 \sqcap_j C_2 \in \mathcal{L}_j^m(z)$. Hence, if we set $\pi(x') = z$, we get that $(\pi(x'), \pi(x)) = (z, \pi(x)) \in m_{ji}$ and $C_1 \sqcap_j C_2 \in \mathcal{L}_j^m(\pi(x'))$. Therefore, Conditions (2.5) are not violated.

- **L \sqcup -rule:** **if** $C_1 \sqcup_i C_2 \in \mathcal{L}_i^g(x)$ and x is not blocked, **then** $C_1 \in \mathcal{L}_i^g(x)$ or $C_2 \in \mathcal{L}_i^g(x)$. In this case, by the induction hypothesis, $C_1 \sqcup_i C_2 \in \mathcal{L}_i^m(\pi(x))$. Thus, by Property (A2), we get that $C_1 \in \mathcal{L}_i^m(\pi(x))$ or $C_2 \in \mathcal{L}_i^m(\pi(x))$, which shows that Conditions (2.5) are not violated.
- **F \sqcup -rule:** **if** $C_1 \sqcup_j C_2 \in \mathcal{L}_i^g(x)$ and x is not blocked, **then** transmit $r^{j \leftarrow i}(x, C_1 \sqcup_j C_2)$, i.e., create $x' \in V_j$, with $x' \in \text{org}(x)$, such that $C_1 \sqcup_j C_2 \in \mathcal{L}_j^g(x')$. In this case, by the induction hypothesis, $C_1 \sqcup_j C_2 \in \mathcal{L}_i^m(\pi(x))$. Thus, by Property (A2), there exists $z \in U_j$, with $(z, \pi(x)) \in m_{ji}$, such that $C_1 \sqcup_j C_2 \in \mathcal{L}_j^m(z)$. Hence, if we set $\pi(x') = z$, we get that $(\pi(x'), \pi(x)) = (z, \pi(x)) \in m_{ji}$ and $C_1 \sqcup_j C_2 \in \mathcal{L}_j^m(\pi(x'))$. Therefore, Conditions (2.5) are not violated.
- **L \forall -rule:** **if** $\forall_i R.C \in \mathcal{L}_i^g(x)$, x is not blocked, and there exists $y \in V_i$, with $R \in \mathcal{L}_i^g(\langle x, y \rangle)$, **then** $C \in \mathcal{L}_i^g(y)$. In this case, by the induction hypothesis, we get that $\forall_i R.C \in \mathcal{L}_i^m(\pi(x))$ and $R \in \mathcal{L}_i^m(\langle \pi(x), \pi(y) \rangle)$, whence, by Property (A3), $C \in \mathcal{L}_i^m(\pi(y))$. Hence, Conditions (2.5) are not violated.
- **F \forall -rule:** **if** $\forall_j R.C \in \mathcal{L}_i^g(x)$ and x is not blocked, **then** transmit $r^{j \leftarrow i}(x, \forall_j R.C)$, i.e., create $x' \in V_j$, with $x' \in \text{org}(x)$, such that $\forall_j R.C \in \mathcal{L}_j^g(x')$. If $\forall_j R.C \in \mathcal{L}_i^g(x)$, then, by the induction hypothesis, $\forall_j R.C \in \mathcal{L}_i^m(x)$, whence, by Property (A3), there exists $z \in U_j$, with $(z, \pi(x)) \in m_{ji}$ such that, for all $w \in U_j$, with $R \in \mathcal{L}_j^m(\langle z, w \rangle)$, $C \in \mathcal{L}_j^m(w)$. Set $\pi(x') = z$. Then we have that $(\pi(x'), \pi(x)) = (z, \pi(x)) \in m_{ji}$ and, by the previous case, $\forall_j R.C \in \mathcal{L}_j^m(\pi(x'))$.
- **L \exists -rule:** **if** $\exists_i R.C \in \mathcal{L}_i^g(x)$, x is not blocked, and there does not exist $y \in V_i$, with $R \in \mathcal{L}_i^g(\langle x, y \rangle)$ and $C \in \mathcal{L}_i^g(y)$, **then** create such a y . In this case, by the induction hypothesis, we get that $\exists_i R.C \in \mathcal{L}_i^m(\pi(x))$, whence, by Property (A4), there exists $z \in U_i$, such that $R \in \mathcal{L}_i^m(\langle \pi(x), z \rangle)$ and $C \in \mathcal{L}_i^m(z)$. Set $\pi(y) = z$. Then $R \in \mathcal{L}_i^m(\langle \pi(x), \pi(y) \rangle)$ and $C \in \mathcal{L}_i^m(\pi(y))$. Hence, Conditions (2.5) are not violated.
- **F \exists -rule:** **if** $\exists_j R.C \in \mathcal{L}_i^g(x)$ and x is not blocked, **then** transmit $r^{j \leftarrow i}(x, \exists_j R.C)$, i.e., create $x' \in V_j$, with $x' \in \text{org}(x)$, such that $\exists_j R.C \in \mathcal{L}_j^g(x')$. If $\exists_j R.C \in \mathcal{L}_i^g(x)$, then,

by the induction hypothesis, $\exists_j R.C \in \mathcal{L}_m^g(x)$, whence, by Property (A4), there exist $z, w \in U_j$, with $(z, \pi(x)) \in m_{ji}$, $R \in \mathcal{L}_j^m(\langle z, w \rangle)$ and $C \in \mathcal{L}_j^m(w)$. Set $\pi(x') = z$. Then we have that $(\pi(x'), \pi(x)) = (z, \pi(x)) \in m_{ji}$ and, by the previous case, $\exists_j R.C \in \mathcal{L}_j^m(\pi(x'))$.

Thus, Conditions (2.5) are not violated in this case either.

G must be clash-free, since, if there existed i, x, C , such that $\{C, \neg_i C\} \subseteq \mathcal{L}_i^g(x)$, then, by Conditions (2.5), $\{C, \neg_i C\} \subseteq \mathcal{L}_i^m(\pi(x))$, which would contradict tableau Property (N1) for M . Hence, whenever an expansion rule is applicable to G , it can be applied in such a way that maintains Conditions (2.5). By Lemma 31, any sequence of rule applications must terminate. Hence, we will obtain a complete and clash-free completion graph G for D from M . ■

By combining Theorems 32, 33 and 34, we obtain the following theorem, which is the main result of the paper.

Theorem 35 *Let Σ be an F- \mathcal{ALCI} ontology and $D \in \widehat{\mathcal{C}}_w$. The F- \mathcal{ALCI} tableau algorithm is a sound, complete, and terminating decision procedure for satisfiability of D as witnessed by T_w . This decision procedure is in 2NEXPTIME w.r.t. the size of D and the sum of the sizes of the modules in $\{T_i\}_{i \in V_w}$.*

2.7 Summary and Discussion

Many semantic web applications require support for knowledge representation and inference over a federation of multiple autonomous ontology modules, without having to combine them in one location. Federated \mathcal{ALCI} or F- \mathcal{ALCI} is a modular description logic, each of whose modules is roughly an \mathcal{ALCI} ontology. F- \mathcal{ALCI} supports importing of both concepts and roles across modules as well as contextualized interpretation of logical connectives. We have presented a federated tableau algorithm for deciding satisfiability of a concept expression from a specific module's point of view in F- \mathcal{ALCI} . We have shown that the algorithm is sound and complete and that its worst-case running time is non-deterministic doubly exponential with respect to the size of the input concept and the sum of the sizes of all modules in the federated ontology. From the complexity-theoretic point of view, this is equivalent to being

non-deterministic doubly exponential with respect to the size of the input concept and the size of the largest module in the federated ontology, since the number of modules is assumed to be fixed. In the non-federated case, several tableau-based algorithms with high complexity upper bounds have been optimized to perform well in practice [3]. We are currently in the process of implementing the federated algorithm. Experimentation and further optimizations may lead to a practically useful federated F- \mathcal{ALCT} reasoner.

Chapter3. REASONING WITH F- \mathcal{ALCI} OVER LATTICES

Abstract

The fully contextualized, federated semantic web language F- \mathcal{ALCI} is generalized to allow reasoning over arbitrary certainty lattices. These are complete completely distributive lattices with a negation operation which is order-reversing and involutive. The resulting language, denoted LF- \mathcal{ALCI} , apart from supporting fully contextualized modular reasoning, encompasses reasoning over structures with a wide variety of orderings, including fuzzy reasoning. The work takes after similar work of Sraccia, who pioneered reasoning over lattices for the description logic \mathcal{ALC} .

3.1 Introduction

This paper is a contribution to the ongoing efforts to endow the world wide web with machine interpretable content and machine interoperable resources and services, thus transforming it into the *semantic web* [11]. Knowledge representation and knowledge acquisition in the semantic web are aimed to be performed, at least partially, by machines and, thus, have to be machine-friendly. The most common platform for this machine-oriented knowledge representation are ontologies. They provide both a syntactic and a semantic framework for reasoning with resources and relations between them. Because in a typical web application many agents contribute parts of an ontology that are often partially overlapping, a significant effort in the area of ontologies focuses on what are called modular or federated ontologies [13, 40, 28, 73, 104, 5, 8]. These are ontologies with multiple modules. Each of the modules is typically constructed independently of other modules and possibly stored in a different machine. The semantics of a modular ontology allows for a smooth interaction between the

overlapping parts of these various independently developed modules.

For constructing ontologies, the most commonly used languages are those that form decidable fragments of first-order logic; they are termed description logics and the reader is referred to the introduction [2] for more details. On the other hand, to support modularization, several modular ontology formalisms have been introduced and explored. Examples include distributed description logics (DDL)[13, 40], \mathcal{E} -Connections [28], semantic importing [73], semantic binding [104], and package-based description logics (P-DL) [5, 8]. In all these approaches, several constructs are provided for sharing of knowledge across ontology modules. An alternative approach to knowledge reuse relies on a particular notion of modularity of ontologies based on the notion of *conservative extensions* [41, 24, 23], which allows ontology modules to be interpreted using standard semantics by requiring that they share the same interpretation domain.

This paper focuses on a specific kind of description logic and a specific kind of modular ontology language. We will combine various features of both languages in order to create a novel modular ontology language that will allow us to reason about uncertain or imprecise knowledge on the semantic web.

The description logic that our language will be based on is the language **L- \mathcal{ALCI}** . Its syntax coincides with the syntax of the well-known description logic \mathcal{ALCI} , which allows forming negations, conjunctions, disjunctions, universal and existential role quantifications of concepts and role inversions. **L- \mathcal{ALCI}** and \mathcal{ALCI} differ from each other with respect to their semantics. Whereas reasoning in \mathcal{ALCI} is based on boolean interpretations and, therefore, can accommodate certain (true or false) knowledge, the semantics of **L- \mathcal{ALCI}** provides for reasoning with uncertain information. More precisely, the concept expressions of **L- \mathcal{ALCI}** are interpreted in a complete completely distributive lattice **L** of certainty values having a negation operation. A description logic along these lines was introduced by Straccia in [93], whose paper was the inspiration for considering this framework. Knowledge management of uncertain or imprecise information has been considered before, for instance in [47, 53, 64] using probability theory, [49] using possibility theory, [75, 90] using many-valued logics and [12, 48, 91, 92, 96, 103] using fuzzy logic. This list of references is indicative of some of the

efforts spent towards handling uncertainty and it is not meant to be exhaustive.

The modular ontology language that we will be basing our investigations on is the language $\mathbf{F}\text{-}\mathcal{ALCI}$, which was introduced and studied in some detail by the authors in [97]. It is a modular ontology language whose main feature is that all its logical connectives are contextualized, i.e., are interpreted locally and their interpretations are then propagated using image-domain relations that relate individuals in different interpretation domains. This language is related to P-DLs, which were previously considered in [5, 8]. P-DLs have only logical negation as a contextualized connective. Furthermore, the semantics in the two platforms are different. P-DL semantics imposes more restrictions on the image-domain relations resulting in a very tight relationship between the overlapping elements of the interpretation domains. Consequently, P-DL semantics is less flexible.

This paper aims at combining the expressive power of $\mathbf{F}\text{-}\mathcal{ALCI}$ with the idea drawn from $\mathbf{L}\text{-}\mathcal{ALCI}$ of allowing interpretations to vary over arbitrary uncertainty lattices. In this way, given an uncertainty lattice \mathbf{L} , a new modular ontology language $\mathbf{LF}\text{-}\mathcal{ALCI}$ is obtained. Its syntax is identical with the syntax of $\mathbf{F}\text{-}\mathcal{ALCI}$. Its semantics, however, allows reasoning with $\mathbf{F}\text{-}\mathcal{ALCI}$ concepts and $\mathbf{F}\text{-}\mathcal{ALCI}$ subsumptions using lattice-theoretic tools. More precisely, membership of an element in a concept extension or of a pair of elements in a role extension is not just true or false but is, instead, assigned a certainty value drawn from the given certainty lattice \mathbf{L} . Based on this basic assignments and various recursive rules involving both the available contextualized logical connectives and the image-domain relations, all memberships in complex concept expressions assume specific certainty values. Apart from formulating this framework, we also present a reduction from $\mathbf{LF}\text{-}\mathcal{ALCI}$ to $\mathbf{L}\text{-}\mathcal{ALCI}$. This reduction allows us to draw conclusions on various computational aspects of $\mathbf{LF}\text{-}\mathcal{ALCI}$ from corresponding statements known about $\mathbf{L}\text{-}\mathcal{ALCI}$. For instance, Straccia [93] has shown that, under appropriate restrictions on the structure of subsumptions and on the certainty lattice \mathbf{L} , satisfiability of an ABox in the language $\mathbf{L}\text{-}\mathcal{ALC}$ is PSPACE-complete with respect to the joint cardinality of the ABox and the lattice. Our result shows that, under the same restrictions as Straccia's on \mathbf{L} , satisfiability in the language $\mathbf{LF}\text{-}\mathcal{ALCI}$ is of the same complexity as satisfiability of an acyclic TBox in

L- \mathcal{ALCI} . The transformation of an acyclic TBox to an ABox, however, may be of exponential length in general. It is still open whether the tracing technique [95] may be used to transform an acyclic **L- \mathcal{ALCI}** TBox to an **L- \mathcal{ALCI}** ABox. In that case, under the same restrictions as Straccia’s on **L**, satisfiability in the language **LF- \mathcal{ALCI}** would also be PSPACE-complete.

In summary, the main contribution of the paper is the study of a modular ontology language that incorporates contextualized and uncertain reasoning. Whereas modularity and contextualization, on the one hand, and uncertainty, on the other, have been studied separately before, to the best of our knowledge, it is the first time that they are being studied in a common framework. Moreover, for acyclic terminological knowledge, the combination of these features does not increase the computational complexity of reasoning.

3.2 A Quick Review of **L- \mathcal{ALCI}** and **F- \mathcal{ALCI}**

3.2.1 **L- \mathcal{ALCI}** Syntax and Semantics

Let $\mathbf{L} = \langle L, \leq \rangle$ be a complete completely distributive lattice, with L its universe and \leq the partial ordering of \mathbf{L} . Denote by \wedge and \vee , as usual, the meet and join operations, respectively, induced by \leq and by 1 and 0 its top and bottom elements. This lattice is perceived as a lattice of “certainty” values into which the expressions of the language **F- \mathcal{ALCI}** will be interpreted. To accommodate negation, we assume that \mathbf{L} is also equipped with a *negation*, i.e., an anti-monotone involutive unary operation \sim with respect to \leq . More explicitly, this means that, for all $a, b \in L$,

- $a \leq b$ implies $\sim b \leq \sim a$ and
- $\sim \sim a = a$.

The term **certainty lattice** is used to refer to the structure $\mathbf{L} = \langle L, \leq, \sim \rangle$. Note that in such a lattice, the De Morgan Laws hold. Examples of certainty lattices, many of which have been widely used in various contexts and for various forms of reasoning in AI, are provided in [93]. Some of them are:

- **Classical 0-1:** The 2-element Boolean algebra $\mathbf{L}_{\{0,1\}}$, where 0 denotes falsity and 1 truth;
- **Fuzzy:** The real unit interval $\mathbf{L}_{[0,1]}$, with negation $\sim \alpha = 1 - \alpha$, for all $\alpha \in [0, 1]$.
- **Four-Valued:** Belnap's *FOUR*, with four values f, t, u, i , satisfying $f \leq u \leq t$ and $f \leq i \leq t$. Negation is given by $\sim f = t$ and $\sim u = u, \sim i = i$. Intuitively, u stands for unknown and i for inconsistency.
- **Many-Valued:** This is the lattice \mathbf{L}_n with n values $0, \frac{1}{n-1}, \dots, \frac{n-2}{n-1}, 1$ with the ordinary ordering. Negation is as in $\mathbf{L}_{[0,1]}$.

The syntax of the language **L- \mathcal{ALCI}** is identical to the syntax of the well-known language \mathcal{ALCI} . More precisely, we have a collection \mathcal{C} of **concept names** and a collection \mathcal{R} or **role names**. Then the set $\widehat{\mathcal{R}}$ or **roles** is the set $\widehat{\mathcal{R}} = \mathcal{R} \cup \mathcal{R}^-$, where $\mathcal{R}^- = \{R^- : R \in \mathcal{R}\}$. The set $\widehat{\mathcal{C}}$ of **concepts** is defined recursively using the following syntax rules for constructing new concepts:

$$A \in \mathcal{C}, \top, \perp, \neg C, C \sqcap D, C \sqcup D, \exists R.C, \forall R.C,$$

for all $C, D \in \widehat{\mathcal{C}}$ and all $R \in \widehat{\mathcal{R}}$. A **(subsumption) formula** is an expression of the form $C \sqsubseteq D$, with $C, D \in \widehat{\mathcal{C}}$. An **ontology** (also known as a **knowledge base** or, for the purposes of this paper, as a **TBox**) is a finite set of formulas. The \mathcal{ALCI} -semantics interprets all logical connectives in the usual way (see Chapter 2 of [2]). This notion of an \mathcal{ALCI} -interpretation is generalized to obtain the notion of an **L- \mathcal{ALCI} -interpretation**, which allows reasoning with uncertain and/or imprecise information based on the certainty values provided by the lattice \mathbf{L} . The definition is essentially that of [93].

An **L- \mathcal{ALCI} -interpretation**, or simply **L-interpretation**, is a pair $\mathcal{I} = \langle \Delta^{\mathcal{I}}, \cdot^{\mathcal{I}} \rangle$, where $\Delta^{\mathcal{I}}$ is a nonempty set, the **domain** of the interpretation, and $\cdot^{\mathcal{I}}$ is an **interpretation function** mapping

- a concept name C into a function $C^{\mathcal{I}} : \Delta^{\mathcal{I}} \rightarrow L$
- a role name R into a function $R^{\mathcal{I}} : \Delta^{\mathcal{I}} \times \Delta^{\mathcal{I}} \rightarrow L$.

For all $a, b \in \Delta^{\mathcal{I}}$, $C^{\mathcal{I}}(a)$ and $R^{\mathcal{I}}(a, b)$ are supposed to provide the degree of certainty of a being an instance of the concept C and of (a, b) being an instance of the role R , respectively, under the interpretation \mathcal{I} . The interpretation function $\cdot^{\mathcal{I}}$ extends to arbitrary roles and concepts by using the following rules recursively, for all $a, b \in \Delta^{\mathcal{I}}$:

$$\begin{aligned}
R^{-\mathcal{I}}(a, b) &= R^{\mathcal{I}}(b, a) \\
\top^{\mathcal{I}}(a) &= 1 \\
\perp^{\mathcal{I}}(a) &= 0 \\
(\neg C)^{\mathcal{I}}(a) &= \sim C^{\mathcal{I}}(a) \\
(C \sqcap D)^{\mathcal{I}}(a) &= C^{\mathcal{I}}(a) \wedge D^{\mathcal{I}}(a) \\
(C \sqcup D)^{\mathcal{I}}(a) &= C^{\mathcal{I}}(a) \vee D^{\mathcal{I}}(a) \\
(\forall R.C)^{\mathcal{I}}(a) &= \bigwedge_{b \in \Delta^{\mathcal{I}}} (\sim R^{\mathcal{I}}(a, b) \vee C^{\mathcal{I}}(b)) \\
(\exists R.C)^{\mathcal{I}}(a) &= \bigvee_{b \in \Delta^{\mathcal{I}}} (R^{\mathcal{I}}(a, b) \wedge C^{\mathcal{I}}(b)).
\end{aligned}$$

Given a formula $C \sqsubseteq D$ and an interpretation \mathcal{I} , \mathcal{I} **satisfies** $C \sqsubseteq D$ or \mathcal{I} is a **model** of $C \sqsubseteq D$ if, for all $a \in \Delta^{\mathcal{I}}$, we have $C^{\mathcal{I}}(a) \leq D^{\mathcal{I}}(a)$. An interpretation \mathcal{I} **satisfies** a knowledge base T or is a **model** of T if it is a model of every formula $\tau \in T$.

Given a specific collection $\mathcal{D} \subseteq \mathcal{C}$ of concept names, an **L-interpretation** \mathcal{I} **\mathcal{D} -satisfies** a knowledge base T or is a **\mathcal{D} -model** of T if it is a model of every formula $\tau \in T$, such that $D^{\mathcal{I}} : \Delta^{\mathcal{I}} \rightarrow \{0, 1\}$, for all $D \in \mathcal{D}$. In other words a \mathcal{D} -model interprets the concept names in \mathcal{D} as subsets of $\Delta^{\mathcal{I}}$ in the ordinary way.

Example: Suppose that we are dealing with a knowledge base T consisting of information about the employees of a certain university. This knowledge base uses \mathcal{ALCI} as the underlying language and contains three concept names **Faculty**, **Highly-Paid-Faculty** and **Productive-Faculty**. Clearly, if the language is to be interpreted in any certainty lattice, the extension of the concept **Faculty** should be a $\{0, 1\}$ -interpretation, whereas the extensions of the other two concept names could be arbitrary evaluations in this lattice. Thus, to answer a query concerning faculty of the university, e.g., to get the certainty values of the set of all faculty that are both highly-paid and productive, we would like to evaluate the extension

of **Highly-Paid-Faculty** \sqcap **Productive-Faculty** in the knowledge base T , that includes the axioms

$$\text{Highly-Paid-Faculty} \sqsubseteq \text{Faculty}$$

$$\text{Productive-Faculty} \sqsubseteq \text{Faculty},$$

under all interpretations that $\{\text{Faculty}\}$ -satisfy T . If there were other concept names in T that should be given “crisp” interpretations, they should be added in this latter set. ■

3.2.2 The Federated Language F- \mathcal{ALCI}

The fully contextualized federated extension of \mathcal{ALCI} , denoted by F- \mathcal{ALCI} , was introduced in [97].

Suppose a directed acyclic graph $G = \langle V, E \rangle$, with $V = \{1, 2, \dots, n\}$, is given. For technical reasons, a loop is added on each vertex of G . For every node $i \in V$, the signature of the i -language includes a set \mathcal{C}_i of i -**concept names** and a set \mathcal{R}_i of i -**role names**. We assume that all sets of names are pairwise disjoint. Out of these, a set of i -**concepts** $\widehat{\mathcal{C}}_i$ and a set of i -**roles** $\widehat{\mathcal{R}}_i$ are built.

Definition 36 (Roles and Concepts) *The set of i -roles or i -role expressions $\widehat{\mathcal{R}}_i$ consists of expressions of the form R, R^- , with $R \in \mathcal{R}_j, (j, i) \in E$.*

The set of i -concepts or i -concept expressions $\widehat{\mathcal{C}}_i$, on the other hand, is defined recursively as follows:

$$A \in \mathcal{C}_j, \top_j, \perp_j, \neg_j C, C \sqcap_j D, C \sqcup_j D, \exists_j R.C, \forall_j R.C, \quad (3.1)$$

where $(j, i) \in E$, $C, D \in \widehat{\mathcal{C}}_i \cap \widehat{\mathcal{C}}_j$ and $R \in \widehat{\mathcal{R}}_i \cap \widehat{\mathcal{R}}_j$.

Using the concepts and roles of F- \mathcal{ALCI} , we define its formulas, as follows:

For any $i \in V$, the i -**formulas** are expressions of the form $C \sqsubseteq D$, with $C, D \in \widehat{\mathcal{C}}_i$. An F- \mathcal{ALCI} -**ontology** or F- \mathcal{ALCI} -**knowledge base** is a collection $T = \{T_i\}_{i \in V}$, where T_i is a finite set of i -formulas, $i \in V$. The T_i 's are referred to as the **modules** of the ontology T .

An F- \mathcal{ALCI} -**interpretation** $\mathcal{I} = \langle \{\mathcal{I}_i\}_{i \in V}, \{r_{ij}\}_{(i,j) \in E} \rangle$ consists of a family $\mathcal{I}_i = \langle \Delta^i, \cdot^i \rangle, i \in V$, of **local interpretations**, together with a family of **image domain relations** $r_{ij} \subseteq$

$\Delta^i \times \Delta^j, (i, j) \in E$, such that $r_{ii} = \text{id}_{\Delta^i}$, for all $i \in V$.

Notation: For a binary relation $r \subseteq \Delta^i \times \Delta^j$, $X \subseteq \Delta^i$ and $S \subseteq \Delta^i \times \Delta^i$, we set

$$r(X) := \{y \in \Delta^j : (\exists x \in X)((x, y) \in r)\},$$

$$r(S) := \{(z, w) \in \Delta^j \times \Delta^j : (\exists (x, y) \in S)((x, z), (y, w) \in r)\}.$$

A local interpretation function \cdot^i interprets i -role names and i -concept names, as well as \perp_i and \top_i , as follows:

- $C^i \subseteq \Delta^i$, for all $C \in \mathcal{C}_i$,
- $R^i \subseteq \Delta^i \times \Delta^i$, for all $R \in \mathcal{R}_i$,
- $\top_i^i = \Delta^i$, $\perp_i^i = \emptyset$.

On the other hand, \cdot^i interprets j -concept names and j -role names, for $(j, i) \in E$, sometimes referred to as **imported** concept and role names, respectively, using the following rules:

- $C^i = r_{ji}(C^j)$, for all $C \in \mathcal{C}_j \cap \widehat{\mathcal{C}}_i$,
- $R^i = r_{ji}(R^j)$, for all $R \in \mathcal{R}_j \cap \widehat{\mathcal{R}}_i$,
- $\top_j^i = r_{ji}(\Delta^j)$, $\perp_j^i = \emptyset$.

The recursive features of the local interpretation function \cdot^i are as follows:

- $R^{-i} = R^{i-}$, for all $R \in \mathcal{R}_i$,
- $(\neg_j C)^i = r_{ji}(\Delta^j \setminus C^j)$
- $(C \sqcap_j D)^i = r_{ji}(C^j \cap D^j)$
- $(C \sqcup_j D)^i = r_{ji}(C^j \cup D^j)$
- $(\exists_j R.C)^i = r_{ji}(\{x \in \Delta^j : (\exists y)((x, y) \in R^j \text{ and } y \in C^j)\})$
- $(\forall_j R.C)^i = r_{ji}(\{x \in \Delta^j : (\forall y)((x, y) \notin R^j \text{ or } y \in C^j)\})$

For all $i \in V$, i -**satisfiability**, denoted by \models_i , is defined by $\mathcal{I} \models_i C \sqsubseteq D$ if $C^i \subseteq D^i$. Given a knowledge base $T = \{T_i\}_{i \in V}$, the interpretation \mathcal{I} is a **model** of T_i , written $\mathcal{I} \models_i T_i$, if $\mathcal{I} \models_i \tau$, for every $\tau \in T_i$. Moreover, \mathcal{I} is a **model** of T , written $\mathcal{I} \models T$, whenever $\mathcal{I} \models_i T_i$, for every $i \in V$.

Let $w \in V$. Define $G_w = \langle V_w, E_w \rangle$ to be the subgraph of G induced by those vertices in G from which w is reachable and $T_w^* := \{T_i\}_{i \in V_w}$. We say that an F- \mathcal{ALCI} -ontology $T = \{T_i\}_{i \in V}$ is **consistent as witnessed by a module** T_w if T_w^* has a model $\mathcal{I} = \langle \{\mathcal{I}_i\}_{i \in V_w}, \{r_{ij}\}_{(i,j) \in E_w} \rangle$, such that $\Delta^w \neq \emptyset$. A concept C is **satisfiable as witnessed by** T_w if there is a model \mathcal{I} of T_w^* , such that $C^w \neq \emptyset$. A concept subsumption $C \sqsubseteq D$ is **valid as witnessed by** T_w , denoted by $C \sqsubseteq_w D$, if, for every model \mathcal{I} of T_w^* , $C^w \subseteq D^w$. An alternative notation for $C \sqsubseteq_w D$ is $T_w^* \models_w C \sqsubseteq D$.

3.3 LF- \mathcal{ALCI} : Reasoning with F- \mathcal{ALCI} over Lattices

We now proceed to describe the syntax and semantics of LF- \mathcal{ALCI} , which extends the modular ontology language F- \mathcal{ALCI} to support reasoning over arbitrary certainty lattices.

Since the syntax of LF- \mathcal{ALCI} is identical with the syntax of F- \mathcal{ALCI} , which was reviewed in the previous section, we concentrate here on the semantics. As contrasted with the language F- \mathcal{ALCI} , the novel feature of the new language is its semantics which allows its expressions to be interpreted as arbitrary values in the certainty lattice \mathbf{L} , rather than just as “true” (1) or “false” (0).

Definition 37 *An interpretation $\mathcal{I} = \langle \{\mathcal{I}_i\}_{i \in V}, \{r_{ij}\}_{(i,j) \in E} \rangle$ consists of a family $\mathcal{I}_i = \langle \Delta^i, \cdot^i \rangle$, $i \in V$, of **local interpretations**, together with a family of **image domain relations** $r_{ij} : \Delta^i \times \Delta^j \rightarrow L$, $(i,j) \in E$, such that, for all $i \in V$,*

$$r_{ii}(a, b) = \begin{cases} 1, & \text{if } a = b \\ 0, & \text{otherwise} \end{cases}$$

A local interpretation function \cdot^i interprets i -role names and i -concept names, as well as

\perp_i and \top_i , as follows:

- $C^i : \Delta^i \rightarrow L$, for all $C \in \mathcal{C}_i$,
- $R^i : \Delta^i \times \Delta^i \rightarrow L$, for all $R \in \mathcal{R}_i$,
- $\top_i^i : \Delta^i \rightarrow L$ is the function $\top_i^i(a) = 1$, for all $a \in \Delta^i$,
- $\perp_i^i : \Delta^i \rightarrow L$ is the function $\perp_i^i(a) = 0$, for all $a \in \Delta^i$.

The interpretations of imported role names and imported concept names are computed by the following rules, for all $a, b \in \Delta^i$:

- $C^i(a) = \bigvee_{c \in \Delta^j} (C^j(c) \wedge r_{ji}(c, a))$, for all $C \in \mathcal{C}_j \cap \widehat{\mathcal{C}}_i$,
- $R^i(a, b) = \bigvee_{c, d \in \Delta^j} (R^j(c, d) \wedge r_{ji}(c, a) \wedge r_{ji}(d, b))$, for all $R \in \mathcal{R}_j \cap \widehat{\mathcal{R}}_i$,
- $\top_j^i(a) = \bigvee_{c \in \Delta^j} (\top_j^j(c) \wedge r_{ji}(c, a)) = \bigvee_{c \in \Delta^j} r_{ji}(c, a)$,
- $\perp_j^i(a) = 0$.

Example: Assume that $\mathbf{L} = \mathbf{L}_{[0,1]}$. Suppose that we are dealing with the knowledge base T containing information about certain products. It consists of two modules T_1 and T_2 describing different but related varieties of products that only partially match (like, e.g., cameras and camcorders). In this context, if the knowledge base is to compute the certainty value of product A in the domain Δ^2 being in the extension of the concept **Expensive**, defined in module T_1 , (see Figure 3.1), it would have to perform the following computation:

$$\begin{aligned}
 \text{Expensive}^2(A) &= (\text{Expensive}^1(a) \wedge r_{12}(a, A)) \vee (\text{Expensive}^1(b) \wedge r_{12}(b, A)) \\
 &= \left(\frac{3}{8} \wedge \frac{3}{4}\right) \vee \left(\frac{3}{4} \wedge \frac{1}{4}\right) \\
 &= \frac{3}{8} \vee \frac{1}{4} \\
 &= \frac{3}{8}.
 \end{aligned}$$

The conclusion is that product A in Δ^2 is expensive with degree $\frac{3}{8}$. ■

The recursive features of the local interpretation function \cdot^i are given, for all $a, b \in \Delta^i$, by:

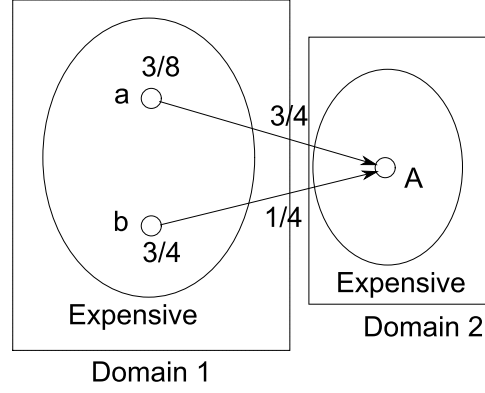
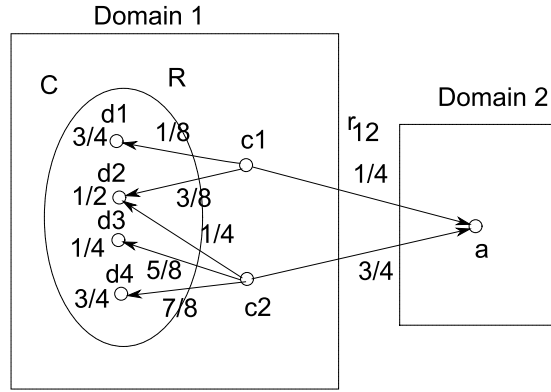


Figure 3.1 An Interpretation over $\mathbf{L}_{[0,1]}$.

- $R^{-i}(a, b) = R^i(b, a)$, for all $R \in \mathcal{R}_i$,
- $(\neg_j C)^i(a) = \bigvee_{c \in \Delta^j} (\sim C^j(c) \wedge r_{ji}(c, a))$
- $(C \sqcap_j D)^i(a) = \bigvee_{c \in \Delta^j} (C^j(c) \wedge D^j(c) \wedge r_{ji}(c, a))$
- $(C \sqcup_j D)^i(a) = \bigvee_{c \in \Delta^j} ((C^j(c) \vee D^j(c)) \wedge r_{ji}(c, a))$
- $(\exists_j R.C)^i(a) = \bigvee_{c \in \Delta^j} (\bigvee_{d \in \Delta^j} (R^j(c, d) \wedge C^j(d)) \wedge r_{ji}(c, a))$
- $(\forall_j R.C)^i(a) = \bigvee_{c \in \Delta^j} (\bigwedge_{d \in \Delta^j} (\sim R^j(c, d) \vee C^j(d)) \wedge r_{ji}(c, a))$

Example: Assume that $\mathbf{L} = \mathbf{L}_{[0,1]}$. We illustrate the application of the recursive \exists -rule by computing $(\exists_1 R.C)^2(a)$ in the interpretation of a federated knowledge base T , that consists of two modules T_1 and T_2 , which is depicted in Figure 3.2. The leftmost column of numbers gives the certainty value of membership in C^1 . The second column provides the certainty values of membership in R^1 and the values between the two rectangular boxes are the certainty values for membership in the image domain relation r_{12} . The computation goes as follows:

Figure 3.2 Another Interpretation over $\mathbf{L}_{[0,1]}$.

$$\begin{aligned}
(\exists_1 R.C)^2(a) &= [[(R^1(c_1, d_1) \wedge C^1(d_1)) \vee (R^1(c_1, d_2) \wedge C^1(d_2))] \wedge r_{12}(c_1, a)] \vee \\
&\quad [[(R^1(c_2, d_2) \wedge C^1(d_2)) \vee (R^1(c_2, d_3) \wedge C^1(d_3))] \vee \\
&\quad (R^1(c_2, d_4) \wedge C^1(d_4))] \wedge r_{12}(c_2, a)] \\
&= [[(\frac{1}{8} \wedge \frac{3}{4}) \vee (\frac{3}{8} \wedge \frac{1}{2})] \wedge \frac{1}{4}] \vee [[(\frac{1}{4} \wedge \frac{1}{2}) \vee (\frac{5}{8} \wedge \frac{1}{4}) \vee (\frac{7}{8} \wedge \frac{3}{4})] \wedge \frac{3}{4}] \\
&= ((\frac{1}{8} \vee \frac{3}{8}) \wedge \frac{1}{4}) \vee ((\frac{1}{4} \vee \frac{1}{4} \vee \frac{3}{4}) \wedge \frac{3}{4}) \\
&= \frac{3}{4}.
\end{aligned}$$

Thus a is an element in $(\exists_1 R.C)^2$ with certainty value $\frac{3}{4}$. ■

For all $i \in V$, **i -satisfiability**, denoted by \models_i , is defined by $\mathcal{I} \models_i C \sqsubseteq D$ if, for all $a \in \Delta^i$, $C^i(a) \leq D^i(a)$. Given a TBox $T = \{T_i\}_{i \in V}$, the interpretation \mathcal{I} is a **model** of T_i , written $\mathcal{I} \models_i T_i$, if $\mathcal{I} \models_i \tau$, for every $\tau \in T_i$. Moreover, \mathcal{I} is a **model** of T , written $\mathcal{I} \models T$, whenever $\mathcal{I} \models_i T_i$, for every $i \in V$.

Let $w \in V$. Define $G_w = \langle V_w, E_w \rangle$ to be the subgraph of G induced by those vertices in G from which w is reachable and $T_w^* := \{T_i\}_{i \in V_w}$. We say that an LF- \mathcal{ALCI} -ontology $T = \{T_i\}_{i \in V}$ is **consistent as witnessed by a module** T_w if T_w^* has a model $\mathcal{I} = \langle \{\mathcal{I}_i\}_{i \in V_w}, \{r_{ij}\}_{(i,j) \in E_w} \rangle$, such that $\Delta^w \neq \emptyset$. A concept C is **satisfiable as witnessed by T_w** if there is a model \mathcal{I} of T_w^* , such that C^w is not the zero function. A concept subsumption $C \sqsubseteq D$ is **valid as witnessed by T_w** , denoted by $C \sqsubseteq_w D$, if, for every model \mathcal{I} of T_w^* , $C^w(a) \leq D^w(a)$, for all $a \in \Delta^w$. An alternative notation for $C \sqsubseteq_w D$ is $T_w^* \models_w C \sqsubseteq D$.

3.4 A Reduction from LF- \mathcal{ALCI} to L- \mathcal{ALCI}

A **reduction** \mathfrak{R} from an LF- \mathcal{ALCI} KB $\Sigma_d = \{T_i\}_{i \in V}$ to an L- \mathcal{ALCI} KB $\Sigma := \mathfrak{R}(\Sigma_d)$ follows along the same lines as a corresponding reduction from F- \mathcal{ALCI} to \mathcal{ALCI} presented in [97] and is obtained as follows:

The signature of Σ is the union of the local signatures of the modules together with a global top \top , a global bottom \perp , local top concepts \top_i , for all $i \in V$, and, finally, a collection of new role names $\{R_{ij}\}_{(i,j) \in E}$, i.e.,

$$\text{Sig}(\Sigma) = \bigcup_i (\mathcal{C}_i \cup \mathcal{R}_i) \cup \{\top, \perp\} \cup \{\top_i : 1 \leq i \leq n\} \cup \{R_{ij} : (i, j) \in E\}.$$

Moreover, various axioms derived from the structure of Σ_d are added to Σ .

- For each $C \in \mathcal{C}_i$, $C \sqsubseteq \top_i$ is added to Σ .
- For each $R \in \mathcal{R}_i$, \top_i is stipulated to be the domain and range of R , i.e., $\top \sqsubseteq \forall R^-. \top_i$ and $\top \sqsubseteq \forall R. \top_i$ are added to Σ .
- For each new role name R_{ij} , \top_i is stipulated to be its domain and \top_j to be its range, i.e., $\top \sqsubseteq \forall R_{ij}^-. \top_i$ and $\top \sqsubseteq \forall R_{ij}. \top_j$ are added to Σ .
- For each $C \sqsubseteq D \in T_i$, $\#_i(C) \sqsubseteq \#_i(D)$ is added to Σ , where $\#_i$ is a function from $\widehat{\mathcal{C}}_i$ to the set of L- \mathcal{ALCI} -concepts. The precise definition of $\#_i$ is given below.

The mapping $\#_i(C)$ serves to maintain the compatibility of the concept domains. It is defined by induction on the structure of $C \in \widehat{\mathcal{C}}_i$:

- $\#_i(C) = C$, if $C \in \mathcal{C}_i$;
- $\#_i(C) = \exists R_{ji}^-. \#_j(C)$, if $C \in \mathcal{C}_j \cap \widehat{\mathcal{C}}_i$;
- $\#_i(\neg_j D) = \exists R_{ji}^-. (\neg \#_j(D) \sqcap \top_j)$;
- $\#_i(D \boxplus_j E) = \exists R_{ji}^-. (\#_j(D) \boxplus \#_j(E))$, where $\boxplus = \sqcap$ or $\boxplus = \sqcup$;
- $\#_i(\exists_j R. D) = \exists R_{ji}^-. (\exists R_{kj}^-. (\exists R. (\exists R_{kj}. \#_j(D))))$, for $R \in \mathcal{R}_k \cup \mathcal{R}_k^-$;

- $\#_i(\forall_j R.D) = \exists R_{ji}^-. (\forall R_{kj}^-. (\forall R. (\forall R_{kj}. \#_j(D))))$, for $R \in \mathcal{R}_k \cup \mathcal{R}_k^-$.

In the next section we show that the reduction \mathfrak{R} is sound and complete in the sense that, if the local top concept \top_w in $\mathfrak{R}(\Sigma_d)$ is $\{\top_i, \perp_i : i \in V\}$ -satisfiable in an **L-ALCI**-model of $\mathfrak{R}(\Sigma_d)$, then Σ_d itself is consistent as witnessed by T_w and conversely. $\{\top_i, \perp_i : i \in V\}$ -satisfiability will be referred to in the sequel as **tb-satisfiability** (top, bottom satisfiability) and a corresponding model termed a **tb-model**.

3.5 Soundness and Completeness of the Reduction \mathfrak{R}

In this section we present the main result of the paper, viz. that the soundness and completeness proofs can be carried out in the case of an interpretation into a general complete completely distributive lattice with negation, rather than just the classical Boolean interpretation.

3.5.1 Soundness

Definition 38 Let $\Sigma_d = \{T_i\}_{i \in V}$ be an **LF-ALCI** KB and $\mathcal{I} = \langle \Delta^{\mathcal{I}}, \cdot^{\mathcal{I}} \rangle$ an interpretation of the **L-ALCI** ontology $\mathfrak{R}(\Sigma_d)$. Construct an interpretation $F(\mathcal{I}) = \langle \{\mathcal{I}_i\}_{i \in V}, \{r_{ij}\}_{(i,j) \in E} \rangle$ for Σ_d as follows:

- $\Delta^i = \{a \in \Delta^{\mathcal{I}} : \top_i^{\mathcal{I}}(a) > 0\}$, for all $i \in V$;
- $C^i(a) = C^{\mathcal{I}}(a)$, for all $a \in \Delta^i$ and every $C \in \mathcal{C}_i$;
- $R^i(a, b) = R^{\mathcal{I}}(a, b)$, for all $a, b \in \Delta^i$ and every $R \in \mathcal{R}_i$;
- $r_{ij}(a, b) = R_{ij}^{\mathcal{I}}(a, b)$, for all $a \in \Delta^i, b \in \Delta^j$ and every $(i, j) \in E$.

We start with an easy technical lemma that shows, roughly speaking, that the image of the interpretation of a concept C under the interpretation of one of the new role names R_{ij} is equal to the interpretation of the concept $\exists R_{ij}^-. C$ in the same model. This lemma is preparatory in dealing with the various cases involved in the definition of the translation function $\#_i$.

Lemma 39 *Let Σ_d be an **LF- \mathcal{ALCI}** KB and $\mathcal{I} = \langle \Delta^{\mathcal{I}}, \cdot^{\mathcal{I}} \rangle$ an interpretation for $\mathfrak{R}(\Sigma_d)$. Then, for every concept $C \in \widehat{\mathcal{C}}_i$,*

$$(\exists R_{ij}^-.C)^{\mathcal{I}}(a) = \bigvee_{c \in \Delta^{\mathcal{I}}} (C^{\mathcal{I}}(c) \wedge R_{ij}^{\mathcal{I}}(c, a)), \text{ for all } a \in \Delta^{\mathcal{I}}.$$

Proof: We do indeed have

$$\begin{aligned} (\exists R_{ij}^-.C)^{\mathcal{I}}(a) &= \bigvee_{c \in \Delta^{\mathcal{I}}} (R_{ij}^{-\mathcal{I}}(a, c) \wedge C^{\mathcal{I}}(c)) \quad (\text{by the definition of } \cdot^{\mathcal{I}}) \\ &= \bigvee_{c \in \Delta^{\mathcal{I}}} (C^{\mathcal{I}}(c) \wedge R_{ij}^{\mathcal{I}}(c, a)) \quad (\text{by the definition of } R_{ij}^{-\mathcal{I}}). \end{aligned}$$

■

Next, we present another technical lemma which supplies the precise value of the interpretation of the concept $\forall R_{kj}^-. (\forall R. (\forall R_{kj}. \#_j(C)))$ in terms of the translation $\#_j(C)$ of a concept $C \in \widehat{\mathcal{C}}_j$ and the role name $R \in \mathcal{R}_k$. This lemma will help us deal with the universal quantification case involved in the recursive definition of the translation function $\#_i$.

Lemma 40 *Let Σ_d be an **LF- \mathcal{ALCI}** KB and $\mathcal{I} = \langle \Delta^{\mathcal{I}}, \cdot^{\mathcal{I}} \rangle$ an interpretation for $\mathfrak{R}(\Sigma_d)$. Then, for all $D \in \widehat{\mathcal{C}}_j$, $R \in \mathcal{R}_k$ and for all $c \in \Delta^{\mathcal{I}}$,*

$$\begin{aligned} (\forall R_{kj}^-. (\forall R. (\forall R_{kj}. \#_j(D))))^{\mathcal{I}}(c) = \\ \bigwedge_{f \in \Delta^{\mathcal{I}}} (\sim \bigvee_{d, e \in \Delta^{\mathcal{I}}} (R_{kj}^{\mathcal{I}}(d, c) \wedge R^{\mathcal{I}}(d, e) \wedge R_{kj}^{\mathcal{I}}(e, f)) \vee \#_j(D)^{\mathcal{I}}(f)). \end{aligned} \tag{3.2}$$

Proof:

Using the definition of $\cdot^{\mathcal{I}}$ three times, complete distributivity of disjunction over infinitary conjunction and the infinitary version of De Morgan's Law, we get

$$(\forall R_{kj}^-. (\forall R. (\forall R_{kj}. \#_j(C))))^{\mathcal{I}}(c)$$

$$\begin{aligned}
&= \bigwedge_{d \in \Delta^{\mathcal{I}}} (\sim R_{kj}^{-\mathcal{I}}(c, d) \vee (\forall R. (\forall R_{kj}. \#_j(D)))^{\mathcal{I}}(d)) \\
&= \bigwedge_{d \in \Delta^{\mathcal{I}}} (\sim R_{kj}^{\mathcal{I}}(d, c) \vee \bigwedge_{e \in \Delta^{\mathcal{I}}} (\sim R^{\mathcal{I}}(d, e) \vee (\forall R_{kj}. \#_j(D))^{\mathcal{I}}(e))) \\
&= \bigwedge_{d \in \Delta^{\mathcal{I}}} (\sim R_{kj}^{\mathcal{I}}(d, c) \vee \bigwedge_{e \in \Delta^{\mathcal{I}}} (\sim R^{\mathcal{I}}(d, e) \vee \\
&\quad \bigwedge_{f \in \Delta^{\mathcal{I}}} (\sim R_{kj}^{\mathcal{I}}(e, f) \vee \#_j(D)^{\mathcal{I}}(f)))) \\
&= \bigwedge_{d \in \Delta^{\mathcal{I}}} (\sim R_{kj}^{\mathcal{I}}(d, c) \vee \bigwedge_{e \in \Delta^{\mathcal{I}}} (\bigwedge_{f \in \Delta^{\mathcal{I}}} (\sim R^{\mathcal{I}}(d, e) \vee \\
&\quad \sim R_{kj}^{\mathcal{I}}(e, f) \vee \#_j(D)^{\mathcal{I}}(f)))) \\
&= \bigwedge_{d \in \Delta^{\mathcal{I}}} (\bigwedge_{e \in \Delta^{\mathcal{I}}} (\bigwedge_{f \in \Delta^{\mathcal{I}}} (\sim R_{kj}^{\mathcal{I}}(d, c) \vee \sim R^{\mathcal{I}}(d, e) \vee \\
&\quad \sim R_{kj}^{\mathcal{I}}(e, f) \vee \#_j(D)^{\mathcal{I}}(f)))) \\
&= \bigwedge_{f \in \Delta^{\mathcal{I}}} (\bigwedge_{d, e \in \Delta^{\mathcal{I}}} (\sim (R_{kj}^{\mathcal{I}}(d, c) \wedge R^{\mathcal{I}}(d, e) \wedge R_{kj}^{\mathcal{I}}(e, f)) \vee \#_j(D)^{\mathcal{I}}(f))) \\
&= \bigwedge_{f \in \Delta^{\mathcal{I}}} (\sim \bigvee_{d, e \in \Delta^{\mathcal{I}}} (R_{kj}^{\mathcal{I}}(d, c) \wedge R^{\mathcal{I}}(d, e) \wedge R_{kj}^{\mathcal{I}}(e, f)) \vee \#_j(D)^{\mathcal{I}}(f)).
\end{aligned}$$

■

To relate the interpretation \mathcal{I} with its federated counterpart $\mathbf{F}(\mathcal{I})$, we need to establish a correspondence between the interpretation of the translation $\#_i(C)$ of a concept $C \in \widehat{\mathcal{C}}_i$ under \mathcal{I} and that of the concept C under $\mathbf{F}(\mathcal{I})$. This relationship is explored in the following lemma.

Lemma 41 *Let Σ_d be an **LF-ALC** KB, $\mathcal{I} = \langle \Delta^{\mathcal{I}}, \cdot^{\mathcal{I}} \rangle$ a tb-interpretation for $\mathfrak{R}(\Sigma_d)$ and $\mathbf{F}(\mathcal{I}) = \langle \{\mathcal{I}_i\}_{i \in V}, \{r_{ij}\}_{(i,j) \in E} \rangle$, with $\mathcal{I}_i = \langle \Delta^i, \cdot^i \rangle$. Then, for all $i \in V$, all $C \in \widehat{\mathcal{C}}_i$ and all $a \in \Delta^i$, $\#_i(C)^{\mathcal{I}}(a) = C^i(a)$.*

Proof: We do this by structural induction on C . Since the De Morgan Laws hold for the Lattice **L**, we will restrict attention to negation, conjunction and value restriction.

For the basis of the induction, if $C \in \mathcal{C}_i, a \in \Delta^i$,

$$\begin{aligned}
\#_i(C)^{\mathcal{I}}(a) &= C^{\mathcal{I}}(a) \quad (\text{by the definition of } \#_i(C)) \\
&= C^i(a), \quad (\text{by the definition of } C^i(a))
\end{aligned}$$

whereas, if $C \in \mathcal{C}_j \cap \widehat{\mathcal{C}}_i$, we get, for all $a \in \Delta^i$,

$$\begin{aligned}
\#_i(C)^{\mathcal{I}}(a) &= (\exists R_{ji}^-. \#_j(C))^{\mathcal{I}}(a) \quad (\text{by the definition of } \#_i(C)) \\
&= \bigvee_{c \in \Delta^{\mathcal{I}}} (\#_j(C)^{\mathcal{I}}(c) \wedge R_{ji}^{\mathcal{I}}(c, a)) \quad (\text{by Lemma 39}) \\
&= \bigvee_{c \in \Delta^{\mathcal{I}}} (C^j(c) \wedge r_{ji}(c, a)) \quad (\text{by the definition of } r_{ji} \\
&\hspace{15em} \text{and the previous case}) \\
&= \bigvee_{c \in \Delta^j} (C^j(c) \wedge r_{ji}(c, a)) \quad (\text{since } C \sqsubseteq \top_j) \\
&= C^i(a). \quad (\text{by the definition of } C^i(a))
\end{aligned}$$

For $C = \neg_j D$, we have, for all $a \in \Delta^i$,

$$\begin{aligned}
\#_i(\neg_j D)^{\mathcal{I}}(a) &= (\exists R_{ji}^-. (\neg \#_j(D) \sqcap \top_j))^{\mathcal{I}}(a) \quad (\text{by the definition of } \#_i(\neg_j D)) \\
&= \bigvee_{c \in \Delta^{\mathcal{I}}} ((\neg \#_j(D) \sqcap \top_j)^{\mathcal{I}}(c) \wedge R_{ji}^{\mathcal{I}}(c, a)) \quad (\text{by Lemma 39}) \\
&= \bigvee_{c \in \Delta^{\mathcal{I}}} ((\neg \#_j(D))^{\mathcal{I}}(c) \wedge \top_j^{\mathcal{I}}(c) \wedge R_{ji}^{\mathcal{I}}(c, a)) \\
&\hspace{1.5em} (\text{by the definition of } \cdot^{\mathcal{I}}) \\
&= \bigvee_{c \in \Delta^{\mathcal{I}}} (\sim \#_j(D)^{\mathcal{I}}(c) \wedge \top_j^{\mathcal{I}}(c) \wedge R_{ji}^{\mathcal{I}}(c, a)) \\
&\hspace{1.5em} (\text{by the definition of } \cdot^{\mathcal{I}}) \\
&= \bigvee_{c \in \Delta^{\mathcal{I}}} (\sim D^j(c) \wedge \top_j^{\mathcal{I}}(c) \wedge r_{ji}(c, a)) \quad (\text{by the definition of } F(\mathcal{I}) \\
&\hspace{15em} \text{and the induction hypothesis}) \\
&= \bigvee_{c \in \Delta^j} (\sim D^j(c) \wedge r_{ji}(c, a)) \quad (\text{lattice-theoretically}) \\
&= (\neg_j D)^i(a). \quad (\text{by the definition of } (\neg_j D)^i(a))
\end{aligned}$$

Note that the fact that \mathcal{I} is a tb-interpretation is crucial in the next-to-last equality. This condition will also be used in the proofs for other connectives.

For \sqcap , we get, for all $a \in \Delta^i$,

$$\begin{aligned}
\#_i(D \sqcap_j E)^{\mathcal{I}}(a) &= (\exists R_{ji}^-. (\#_j(D) \sqcap \#_j(E)))^{\mathcal{I}}(a) \quad (\text{by the definition of } \\
&\quad \#_i(D \sqcap_j E)) \\
&= \bigvee_{c \in \Delta^{\mathcal{I}}} ((\#_j(D) \sqcap \#_j(E))^{\mathcal{I}}(c) \wedge R_{ji}^{\mathcal{I}}(c, a)) \quad (\text{by Lemma 39}) \\
&= \bigvee_{c \in \Delta^{\mathcal{I}}} ((\#_j(D)^{\mathcal{I}}(c) \wedge \#_j(E)^{\mathcal{I}}(c)) \wedge R_{ji}^{\mathcal{I}}(c, a)) \\
&\quad (\text{by the definition of } \cdot^{\mathcal{I}}) \\
&= \bigvee_{c \in \Delta^{\mathcal{I}}} ((D^j(c) \wedge E^j(c)) \wedge r_{ji}(c, a)) \quad (\text{by the definition of } \\
&\quad \mathbf{F}(\mathcal{I}) \text{ and the induction hypothesis}) \\
&= \bigvee_{c \in \Delta^j} ((D^j(c) \wedge E^j(c)) \wedge r_{ji}(c, a)) \quad (\text{since } D^j, E^j \leq \top_j^{\mathcal{I}}) \\
&= (D \sqcap_j E)^i(a). \quad (\text{by the definition of } (D \sqcap_j E)^i(a))
\end{aligned}$$

For $C = \#_i(\forall_j R.D)$, with $R \in \mathcal{R}_k$, we obtain, for all $a \in \Delta^i$,

$$\begin{aligned}
\#_i(\forall_j R.D)^{\mathcal{I}}(a) &= (\exists R_{ji}^-. (\forall R_{kj}^-. (\forall R. (\forall R_{kj}. (\#_j(D))))))^{\mathcal{I}}(a) \\
&\quad (\text{by the definition of } \#_i(\forall_j R.D)) \\
&= \bigvee_{c \in \Delta^{\mathcal{I}}} ((\forall R_{kj}^-. (\forall R. (\forall R_{kj}. (\#_j(D))))^{\mathcal{I}}(c) \wedge R_{ji}^{\mathcal{I}}(c, a)) \\
&\quad (\text{by Lemma 39}) \\
&= \bigvee_{c \in \Delta^j} ((\forall R_{kj}^-. (\forall R. (\forall R_{kj}. (\#_j(D))))^{\mathcal{I}}(c) \wedge r_{ji}(c, a)) \\
&= \bigvee_{c \in \Delta^j} (\bigwedge_{f \in \Delta^{\mathcal{I}}} (\sim \bigvee_{d, e \in \Delta^{\mathcal{I}}} (R_{kj}^{\mathcal{I}}(d, c) \wedge R^{\mathcal{I}}(d, e) \wedge R_{kj}^{\mathcal{I}}(e, f)) \vee \\
&\quad \#_j(D)^{\mathcal{I}}(f)) \wedge r_{ji}(c, a)) \quad (\text{by Equation (3.2)}) \\
&= \bigvee_{c \in \Delta^j} (\bigwedge_{f \in \Delta^j} (\sim \bigvee_{d, e \in \Delta^k} (r_{kj}(d, c) \wedge R^k(d, e) \wedge r_{kj}(e, f)) \vee \\
&\quad D^j(f)) \wedge r_{ji}(c, a)) \\
&\quad (\text{by the definition of } \mathbf{F}(\mathcal{I}) \text{ and the induction hypothesis}) \\
&= \bigvee_{c \in \Delta^j} (\bigwedge_{f \in \Delta^j} (\sim R^j(c, f) \vee D^j(f)) \wedge r_{ji}(c, a)) \\
&\quad (\text{by the definition of } R^j(c, f)) \\
&= (\forall_j R.D)^i(a). \quad (\text{by the definition of } (\forall_j R.D)^i(a))
\end{aligned}$$

■

The following is the soundness theorem for the reduction \mathfrak{R} .

Theorem 42 (Soundness) *Let Σ_d be an **LF- \mathcal{ALCI}** KB, and T_w a module of Σ_d . If \top_w is tb-satisfiable with respect to $\mathfrak{R}(T_w^*)$, then Σ_d is consistent as witnessed by T_w .*

Proof:

Suppose that \top_w is tb-satisfiable with respect to $\mathfrak{R}(T_w^*)$. Then $\mathfrak{R}(T_w^*)$ has a model $\mathcal{I} = \langle \Delta^{\mathcal{I}}, \cdot^{\mathcal{I}} \rangle$, such that $\top_w^{\mathcal{I}}$ is not identically 0. Our goal is to show that $\mathbf{F}(\mathcal{I}) = \langle \{\mathcal{I}_i\}_{i \in V_w}, \{r_{ij}\}_{(i,j) \in E_w} \rangle$ is a model of T_w^* , such that $\Delta^w \neq \emptyset$.

Clearly, we have $\Delta^w = \{a \in \Delta^{\mathcal{I}} : \top_w^{\mathcal{I}}(a) \neq 0\} \neq \emptyset$, by the hypothesis. So it suffices to show that $\mathbf{F}(\mathcal{I})$ is a model of the federated ontology T_w^* , i.e., that it satisfies $\mathcal{I}_i \models T_i$, for every $i \in V_w$. Suppose that $C \sqsubseteq D \in T_i$. By the construction of $\mathfrak{R}(T_w^*)$ and the fact that $\mathcal{I} \models \mathfrak{R}(T_w^*)$, we must have, for all $a \in \Delta^{\mathcal{I}}$, $\#_i(C)^{\mathcal{I}}(a) \leq \#_i(D)^{\mathcal{I}}(a)$, whence, by Lemma 41, for all $a \in \Delta^i$, $C^i(a) \leq D^i(a)$, showing that $\mathbf{F}(\mathcal{I}) \models T_w^*$. \blacksquare

3.5.2 Completeness of the Reduction

We turn now to the proof of the completeness of the reduction \mathfrak{R} . Informally speaking, it will be shown that, if an **LF- \mathcal{ALCI}** KB Σ_d is consistent as witnessed by a module T_w , then the corresponding local top concept \top_w in $\Sigma = \mathfrak{R}(\Sigma_d)$ is satisfiable.

Definition 43 *Suppose that Σ_d is an **LF- \mathcal{ALCI}** KB and that $\mathcal{I}_d = \langle \{\mathcal{I}_i\}_{i \in V}, \{r_{ij}\}_{(i,j) \in E} \rangle$ is a model of Σ_d . Construct a tb-interpretation $\mathcal{I} := \mathbf{G}(\mathcal{I}_d) = \langle \Delta^{\mathcal{I}}, \cdot^{\mathcal{I}} \rangle$ of $\mathfrak{R}(\Sigma_d)$ as follows:*

- $\Delta^{\mathcal{I}} = \bigcup_{i \in V} \Delta^i$;
- $\top_i^{\mathcal{I}}(a) = \begin{cases} 1, & \text{if } a \in \Delta^i \\ 0, & \text{otherwise} \end{cases}, \quad \text{for every } a \in \Delta^{\mathcal{I}}, i \in V$;
- $C^{\mathcal{I}}(a) = \begin{cases} C^i(a), & \text{if } a \in \Delta^i \\ 0, & \text{otherwise} \end{cases}, \quad \text{for every } a \in \Delta^{\mathcal{I}}, C \in \mathcal{C}_i$;
- $R^{\mathcal{I}}(a, b) = \begin{cases} R^i(a, b), & \text{if } a, b \in \Delta^i \\ 0, & \text{otherwise} \end{cases}, \quad \text{for every } a, b \in \Delta^{\mathcal{I}}, R \in \mathcal{R}_i$;

$$\bullet R_{ij}^{\mathcal{I}}(a, b) = \begin{cases} r_{ij}(a, b), & \text{if } (a, b) \in \Delta^i \times \Delta^j \\ 0, & \text{otherwise} \end{cases}, \quad \text{for every } a, b \in \Delta^{\mathcal{I}}, (i, j) \in E.$$

To relate the federated interpretation \mathcal{I}_d with its single module-counterpart $\mathcal{I} := \mathbf{G}(\mathcal{I}_d)$, we need to establish a correspondence between the interpretation of the translation $\#_i(C)$ of a concept $C \in \widehat{\mathcal{C}}_i$ under $\mathbf{F}(\mathcal{I}_d)$ and that of the concept C under \mathcal{I} . Such a correspondence is established in the following lemma.

Lemma 44 *Let Σ_d be an **LF-ALCI** KB, $\mathcal{I}_d = \langle \{\mathcal{I}_i\}_{i \in V}, \{r_{ij}\}_{(i,j) \in E} \rangle$ a model of Σ_d and set $\mathcal{I} := \mathbf{G}(\mathcal{I}_d) = \langle \Delta^{\mathcal{I}}, \cdot^{\mathcal{I}} \rangle$. Then, for all $i \in V$, $C \in \widehat{\mathcal{C}}_i$ and $a \in \Delta^i$, $\#_i(C)^{\mathcal{I}}(a) = C^i(a)$.*

Proof:

This will follow directly from Lemma 41 once it is shown that $\mathcal{I}_d = \mathbf{F}(\mathbf{G}(\mathcal{I}_d))$. We have, using the full model names to keep notation clear,

- For all $i \in V$, $\Delta^{\mathbf{F}(\mathbf{G}(\mathcal{I}_d))i} = \{a \in \Delta^{\mathbf{G}(\mathcal{I}_d)} : \top_i^{\mathbf{G}(\mathcal{I}_d)}(a) \neq 0\} = \Delta^{(\mathcal{I}_d)i}$.
- For every $C \in \mathcal{C}_i$ and all $a \in \Delta^i$, $C^{\mathbf{F}(\mathbf{G}(\mathcal{I}_d))i}(a) = C^{\mathbf{G}(\mathcal{I}_d)}(a) = C^{(\mathcal{I}_d)i}(a)$.
- For all $R \in \mathcal{R}_i$, $a, b \in \Delta^i$, we get $R^{\mathbf{F}(\mathbf{G}(\mathcal{I}_d))i}(a, b) = R^{\mathbf{G}(\mathcal{I}_d)}(a, b) = R^{(\mathcal{I}_d)i}(a, b)$.
- For all $(i, j) \in E$, $a \in \Delta^i$, $b \in \Delta^j$, $r_{ij}^{\mathbf{F}(\mathbf{G}(\mathcal{I}_d))}(a, b) = R_{ij}^{\mathbf{G}(\mathcal{I}_d)}(a, b) = r_{ij}^{\mathcal{I}_d}(a, b) = r_{ij}(a, b)$,
where the superscripts of r_{ij} 's specify the model of which they are part.

Therefore, we do indeed have $\mathcal{I}_d = \mathbf{F}(\mathbf{G}(\mathcal{I}_d))$. By Lemma 41, for all $a \in \Delta^i$, $\#_i(C)^{\mathcal{I}}(a) = \#_i(C)^{\mathbf{G}(\mathcal{I}_d)}(a) = C^{\mathbf{F}(\mathbf{G}(\mathcal{I}_d))i}(a) = C^{(\mathcal{I}_d)i}(a) = C^i(a)$. ■

The main goal of this section is to show that the converse of Theorem 42 also holds.

Theorem 45 (Completeness) *Let $\Sigma_d = \{T_i\}_{i \in V}$ be an **LF-ALCI** ontology. If Σ_d is consistent as witnessed by a module T_w , then \top_w is tb-satisfiable with respect to $\mathfrak{R}(T_w^*)$.*

Proof:

Suppose that Σ_d is consistent as witnessed by T_w . Thus, it has a model $\mathcal{I}_d = \langle \{\mathcal{I}_i\}_{i \in V}, \{r_{ij}\}_{(i,j) \in E} \rangle$, such that $\Delta^w \neq \emptyset$. We proceed to show that $\mathcal{I} := \mathbf{G}(\mathcal{I}_d)$ is a tb-model of $\mathfrak{R}(T_w^*)$, such that $\top_w^{\mathcal{I}}$ is not identically 0.

Since, by hypothesis, $\Delta^w \neq \emptyset$, there exists $a \in \Delta^w$. Thus, by the definition of $\top_w^{\mathcal{I}}$, $\top_w^{\mathcal{I}}(a) = 1$ and, therefore, $\top_w^{\mathcal{I}}$ is not identically 0.

Clearly, if $C \in \mathcal{C}_i$, then, for all $a \in \Delta^{\mathcal{I}}$,

$$C^{\mathcal{I}}(a) = \left\{ \begin{array}{ll} C^i(a), & \text{if } a \in \Delta^i \\ 0, & \text{otherwise} \end{array} \right\} \leq \left\{ \begin{array}{ll} 1, & \text{if } a \in \Delta^i \\ 0, & \text{otherwise} \end{array} \right\} = \top_i^{\mathcal{I}}(a),$$

whence $C \sqsubseteq \top_i$ holds in \mathcal{I} .

To see that $\top \sqsubseteq \forall R^-. \top_i$ holds in \mathcal{I} , we must show that, for all $a \in \Delta^{\mathcal{I}}$, $\top^{\mathcal{I}}(a) \leq \bigwedge_{c \in \Delta^{\mathcal{I}}} (\sim R^{\mathcal{I}}(c, a) \vee \top_i^{\mathcal{I}}(c))$. In turn, it suffices to show that, for all $c \in \Delta^{\mathcal{I}}$, $\sim R^{\mathcal{I}}(c, a) \vee \top_i^{\mathcal{I}}(c) = 1$. In fact, if $c \notin \Delta^i$, then $R^{\mathcal{I}}(c, a) = 0$, whence $\sim R^{\mathcal{I}}(c, a) = 1$ and, thus, $\sim R^{\mathcal{I}}(c, a) \vee \top_i^{\mathcal{I}}(c) = 1$. If, on the other hand, $c \in \Delta^i$, then $\top_i^{\mathcal{I}}(c) = 1$, whence, again, $\sim R^{\mathcal{I}}(c, a) \vee \top_i^{\mathcal{I}}(c) = 1$. The fact that $\mathcal{I} \models \top \sqsubseteq \forall R. \top_i$ may be shown similarly. Also along the same lines follow the proofs that the two concept inclusion axioms $\top \sqsubseteq \forall R_{ij}^-. \top_i$ and $\top \sqsubseteq \forall R_{ij}. \top_j$ are valid in \mathcal{I} .

Finally, suppose that $\#_i(C) \sqsubseteq \#_i(D)$ is in $\mathfrak{R}(\Sigma_d)$. Then $C \sqsubseteq D \in T_i$ and, since $\mathcal{I}_d \models \Sigma_d$, we must have, for all $a \in \Delta^i$, $C^i(a) \leq D^i(a)$. Therefore, by Lemma 44, for all $a \in \Delta^{\mathcal{I}}$, $\#_i(C)^{\mathcal{I}}(a) \leq \#_i(D)^{\mathcal{I}}(a)$, which shows that $\mathcal{I} \models \#_i(C) \sqsubseteq \#_i(D)$. Thus, if $\mathcal{I}_d \models T_w^*$, we must have that $\mathfrak{G}(\mathcal{I}_d) \models \mathfrak{R}(T_w^*)$. This concludes the proof that, if Σ_d is consistent as witnessed by a package T_w , then \top_w is tb-satisfiable with respect to $\mathfrak{R}(T_w^*)$. ■

By combining Theorems 42 and 45 we get the following

Theorem 46 (Soundness and Completeness) *Let $\Sigma_d = \{T_i\}_{i \in V}$ be an **LF-ALCI** ontology. Σ_d is consistent as witnessed by a module T_w if and only if \top_w is tb-satisfiable with respect to $\mathfrak{R}(T_w^*)$.*

3.5.3 Complexity

Straccia [93] provides a tableau-style calculus for deciding satisfiability in the language **L-ALC** under some restrictions on both the form of the TBox and the certainty lattice **L**.

In the TBox, subsumptions are restricted to two kinds of axioms: concept specializations and concept definitions. A concept specialization is an axiom of the form $A \sqsubseteq C$ and a concept

definition an axiom of the form $A = C$, where, in both cases, A is a concept name and C an arbitrary concept expression. In the TBox no concept name appears more than once on the left hand side of an axiom and no cyclic definitions are allowed.

The certainty lattice \mathbf{L} is assumed to be **ps-safe** (polynomial space safe). This requires that \mathbf{L} is finite and that the decision problem whether a set of constraints is inconsistent is in polynomial space w.r.t. *combined complexity*. This is the complexity with respect to the sum of the cardinalities of the knowledge base and of the certainty lattice. Using a tableau-like calculus, he shows that satisfiability of an $\mathbf{L}\text{-}\mathcal{ALC}$ knowledge base, having a TBox that satisfies the restrictions listed above, is PSPACE-complete with respect to the combined complexity of a ps-safe lattice, provided that the technique of transforming the TBox into an ABox [71] does not cause exponential blow up [93, 71]. It is not difficult to see that, as in the case of \mathcal{ALCI} versus \mathcal{ALC} , Straccia's algorithm may be extended to accommodate inverse roles, i.e., to the language $\mathbf{L}\text{-}\mathcal{ALCI}$, while preserving the complexity.

To be able to use Straccia's result to obtain PSPACE complexity for the consistency problem for $\mathbf{LF}\text{-}\mathcal{ALCI}$, apart from imposing similar restrictions to Straccia's on the certainty lattice and the form of our local TBoxes, it is necessary that transforming the $\mathbf{L}\text{-}\mathcal{ALCI}$ TBox resulting from the reduction \mathfrak{R} into an ABox does not cause exponential blowup. Since this cannot be generally ensured, we define a federated $\mathbf{LF}\text{-}\mathcal{ALCI}$ terminological knowledge base $\Sigma_d = \{T_i\}_{i \in V}$ to be **tame** if satisfiability of $\mathfrak{R}(\Sigma_d)$ can be reduced to that of an ABox by the technique of [71] in polynomial space.

Example: Assume that we have available two TBoxes T_1 and T_2 . In T_1 , two concept names A, B are defined, together with one role name R , which are related by the subsumption

$$A \sqsubseteq \exists_1 R.B.$$

Tbox T_2 imports concepts A, B and the role R from T_1 and has no local (concept or role) names. Let us briefly exemplify the process that would be followed, using the algorithm of Straccia [93], to check whether it holds from the point of view of T_2 that $A \sqsubseteq \exists_2 R.B$.

First, the ontology is transformed using \mathfrak{R} to the following single-module ontology:

$$\begin{aligned}
A &\sqsubseteq \top_1 \\
B &\sqsubseteq \top_1 \\
\top &\sqsubseteq \forall R^-. \top_1 \\
\top &\sqsubseteq \forall R. \top_1 \\
\top &\sqsubseteq \forall R_{12}^-. \top_1 \\
\top &\sqsubseteq \forall R_{12}. \top_2 \\
A &\sqsubseteq \exists R. B
\end{aligned}$$

To conform to the requirements imposed by Straccia, we introduce two new concept names A' and B' and convert this ontology to the following:

$$\begin{aligned}
A &= \top_1 \sqcap \exists R. B \sqcap A' \\
B &= \top_1 \sqcap B' \\
\top &= \forall R^-. \top_1 \sqcap \forall R. \top_1 \sqcap \forall R_{12}^-. \top_1 \sqcap \forall R_{12}. \top_2
\end{aligned}$$

which, by substituting $\top_1 \sqcap B'$ for B , gives the following ontology involving only the undefined concepts \top_1, \top_2, A' and B' :

$$\begin{aligned}
A &= \top_1 \sqcap \exists R. (\top_1 \sqcap B') \sqcap A' \\
B &= \top_1 \sqcap B' \\
\top &= \forall R^-. \top_1 \sqcap \forall R. \top_1 \sqcap \forall R_{12}^-. \top_1 \sqcap \forall R_{12}. \top_2
\end{aligned}$$

The query asks whether $A \sqsubseteq_2 \exists R. B$. This is converted to a single-module query by applying \mathfrak{R} :

$$\exists R_{12}^-. A \sqsubseteq \exists R_{12}^-. \exists R. \exists R_{12}. \exists R_{12}^-. B.$$

Finally, we need to apply Straccia's algorithm to check whether

$$\langle (a : \top \sqcap \exists R_{12}^-. A \sqcap \neg \exists R_{12}^-. \exists R. \exists R_{12}. \exists R_{12}^-. B) \not\leq 0 \rangle$$

is satisfiable. In that case, of course, the required subsumption will not be valid in the given ontology. This last test takes the following form after performing all necessary substitutions:

$$\begin{aligned} & \langle (a : \forall R^-. \top_1 \sqcap \forall R. \top_1 \sqcap \forall R_{12}^-. \top_1 \sqcap \forall R_{12}. \top_2 \\ & \quad \sqcap \exists R_{12}^-. (\top_1 \sqcap \exists R. (\top_1 \sqcap B') \sqcap A') \\ & \quad \sqcap \neg \exists R_{12}^-. \exists R. \exists R_{12}. \exists R_{12}^-. (\top_1 \sqcap B')) \not\leq 0 \rangle \end{aligned}$$

■

Assuming that the certainty lattice we reason with is ps-safe and that the terminological axioms used in each module satisfy the same restrictions imposed by Straccia in [93] and that, furthermore, our federated knowledge base is tame, Straccia's result (Proposition 4 of [93]), combined with Theorem 46, imply that satisfiability in **L- \mathcal{ALCI}** is also PSPACE-complete with respect to the combined complexity of the ps-safe lattice.

3.6 Summary

In this paper we have introduced a new modular ontology language, **LF- \mathcal{ALCI}** , that allows uncertain or imprecise fully contextualized reasoning in a federated setting. An **LF- \mathcal{ALCI}** ontology consists of multiple ontology modules each of which can be viewed as an \mathcal{ALCI} ontology. The interpretations of memberships in concept and role extensions are taking values in a certainty lattice **L**. This is a complete completely distributive lattice with a negation operation. Concept and role names can be shared by “importing” relations among modules, which are also interpreted in an uncertain fashion.

A reduction is provided from the federated language **LF- \mathcal{ALCI}** to the language **L- \mathcal{ALCI}** . This language is very similar to one considered in [93]. Using techniques very similar to the ones employed in [93], it may be shown that, under certain restrictions on **L** and the form of subsumptions allowed in the knowledge base, satisfiability in **L- \mathcal{ALCI}** is PSPACE-complete with respect to the joint cardinality of the knowledge base and the lattice. Our reduction entails the PSPACE-completeness of the satisfiability problem in **LF- \mathcal{ALCI}** subject to the same restrictions on the lattice **L** and the form of subsumptions allowed in the local modules.

Chapter 4. PROBABILISTIC FEDERATED \mathcal{ALCI}

Abstract

In previous work, we have introduced a fully contextualized federated ontology language F- \mathcal{ALCI} , based on the well-known description logic \mathcal{ALCI} . Inspired by the work of Lukasiewicz on expressive probabilistic logics, we augment that work by considering a probabilistic extension of F- \mathcal{ALCI} , termed PF- \mathcal{ALCI} . Although its modules employ a less expressive description logic than the rich $\mathcal{SHIF}(\mathbf{D})$ or $\mathcal{SHOIN}(\mathbf{D})$ of Lukasiewicz and, in particular, do not provide support for concrete domains, PF- \mathcal{ALCI} is the first ontology language in the literature to offer modularity and contextualization of all logical connectives combined with the ability to express probabilistic terminological and default knowledge.

4.1 Introduction

The large amount of data and services that have become available on the world-wide web have led to the *semantic web initiative* [11, 36], which aims at making information machine-interpretable and services machine-operable so that data discovery, integration and navigation can be enhanced. The precision in the definition and the meaning of the terms representing the available information, required for succeeding in this goal, is provided by organizing them into ontologies. Ontologies are knowledge bases that typically cover a specific domain of expertise. Different ontologies may cover related domains with partially overlapping, or interdependent, information, but are typically developed independently of each other. One of the most widely used languages for ontology construction is OWL [86]. Ontology languages are based on description logics [2], which, typically, are decidable fragments of first-order logic or various other decidable extensions with additional constructs that are used to enrich expressivity without

compromising decidability [16]. The basic building blocks of a description logic are concepts and roles. Concepts represent classes of individuals in the domain of discourse and roles represent relationships between individuals. The most elementary statements that are encoded in a description logic knowledge base and on which we will focus in this paper are subsumption relationships between concepts.

In various applications of description logics in the semantic web, the need arises to express uncertain or imprecise information and to reason about it. In this direction a body of work has focused on integrating fuzzy representation and reasoning into ontology languages (see, e.g., [91, 92, 93]). On the other hand, an alternative approach is to use probabilistic methods to represent and reason about uncertain information on the web. This has been explored extensively in the area of logic programs (see, e.g., [62]) and various researchers have advocated and treated the introduction of probabilistic features in knowledge representation [47, 53, 54, 55, 64] and ontology engineering [33, 34, 102, 77, 37, 72, 30, 31].

Typically, development of ontologies in the semantic web is occurring autonomously by independent contributors, each of whom addresses a different area of expertise. But the ontology modules that are constructed in this federated fashion are not entirely disjoint. They may cover related or partially overlapping domains, e.g., biology, medicine, pharmacology. In order to avoid reconstructing the same terminology and repeating parts of an already existing ontology, tools have been developed that allow an ontology developer to reuse concepts and definitions from other ontology modules. The theoretical study on the foundations of ontology languages that allow this feature has led to the development of several possible platforms that may be used for selectively reusing parts of other ontology modules in the development of a new ontology. These modular ontology languages include distributive description logics [14, 40], \mathcal{E} -connections [28], semantic importing [73], semantic binding [104] and package-based description logics [5, 8]. A slightly different approach that also has as its main goal partial reuse of available knowledge is based on the notion of conservative extensions [41, 24, 23]. Of particular interest to us, since it will form the foundation for our studies in this paper, is the framework of federated, fully-contextualized description logics that was introduced recently in

[97, 98]. Apart from enabling the user to partially reuse information by importing concepts and roles from different modules, it also recognizes the need to contextualize information. This need arises because imported terms from other modules may be interpreted differently depending on the context in which they are being reused. Context as a key concept in reasoning in AI has been studied before in [20, 21] and, more specifically, in the area of ontology languages in [15, 38, 39]. The additional recognition of the need to reason with imprecise or fuzzy information in this federated setting has recently led to the formulation of a federated reasoning framework [100], where instead of a two-valued semantics, an arbitrary certainty lattice may be used, as was done previously in the single-module setting in [93].

In the present work, we introduce probabilistic terminological axioms in the federated fully-contextualized description logic $F\text{-}\mathcal{ALCI}$ to obtain the probabilistic federated description logic $PF\text{-}\mathcal{ALCI}$. We follow in this endeavor the leads from the pioneering work of Lukasiewicz [64], where probabilistic analogs of the very expressive description logics $\mathcal{SHIF}(\mathbf{D})$ and $\mathcal{SHOIN}(\mathbf{D})$ were introduced and studied in detail. Because this is, to the best of our knowledge, the first attempt at the creation of a relatively expressive *modular* description logic with *probabilistic features*, we opted for a rather simplified version of the description logic used, as compared with the powerful logics used in [64]. More expressive DLs and a more general framework will be studied in future work. Our framework has the following three limitations when comparing the underlying language used with those of Lukasiewicz: First, \mathcal{ALCI} is significantly less expressive than either $\mathcal{SHIF}(\mathbf{D})$ or $\mathcal{SHOIN}(\mathbf{D})$. Second, we do not treat concrete domains as does Lukasiewicz. Finally, we restrict our attention only to terminological axioms. Despite these simplifications our innovation relies on several features that are introduced *collectively* for the first time in an ontology language. First, our language is *modular*. That is, its semantics handles readily interactions between various modules that are developed independently on the web. Second, in each of these modules, all logical connectives are *contextualized*. Each logical connective has a local meaning that is transferred across modules via image domain relations. Finally, several of the nice *probabilistic features* of Lukasiewicz’s approach pertaining to default and probabilistic terminological axioms still hold

in the distributed context, despite the limited expressivity of the underlying description logic.

4.2 A Quick Review of \mathcal{ALCI} and $\mathbf{F}\text{-}\mathcal{ALCI}$

4.2.1 \mathcal{ALCI} Basics

Recall, e.g., from [2], that the description logic \mathcal{ALCI} consists of role expressions and concept expressions that are built starting from two disjoint collections of concept names \mathcal{C} and role names \mathcal{R} , using the top and bottom concepts, negation, conjunction, disjunction, value and existential restriction (for concepts) and inverse roles. More precisely, if A is a concept name and C, D are concept expressions, then

$$\top, \perp, A, \neg C, C \sqcap D, C \sqcup D, \forall R.C \text{ and } \exists R.C$$

are concept expressions, where R is a role expression, i.e., a role name R or of the form R^- , with R a role name. The set of role expressions is denoted by $\widehat{\mathcal{R}}$ and the set of concept expressions by $\widehat{\mathcal{C}}$. A subsumption in \mathcal{ALCI} is a formula of the form $C \sqsubseteq D$, where $C, D \in \widehat{\mathcal{C}}$. An ontology T (also known as knowledge base (KB) or TBox) is a finite set of subsumptions. This language is provided a formal semantics as follows: An interpretation for T is a pair $\mathcal{I} = \langle \Delta^{\mathcal{I}}, \cdot^{\mathcal{I}} \rangle$, where $\Delta^{\mathcal{I}}$ is a nonempty set, called the domain of the interpretation, and $\cdot^{\mathcal{I}}$ is a function, that assigns to each concept name C a set $C^{\mathcal{I}} \subseteq \Delta^{\mathcal{I}}$ and to each role name R a set $R^{\mathcal{I}} \subseteq \Delta^{\mathcal{I}} \times \Delta^{\mathcal{I}}$, such that $\top^{\mathcal{I}} = \Delta^{\mathcal{I}}$ and $\perp^{\mathcal{I}} = \emptyset$. One uses the recursive nature of the concept expressions to extend the function $\cdot^{\mathcal{I}}$ over all role and concept expressions as follows: Let $(R^-)^{\mathcal{I}} = (R^{\mathcal{I}})^-$, the inverse relation of $R^{\mathcal{I}}$, for every $R \in \mathcal{R}$, and, for all concept expressions C, D and role expressions R ,

- $(\neg C)^{\mathcal{I}} = \Delta^{\mathcal{I}} \setminus C^{\mathcal{I}}$;
- $(C \sqcap D)^{\mathcal{I}} = C^{\mathcal{I}} \cap D^{\mathcal{I}}$;
- $(C \sqcup D)^{\mathcal{I}} = C^{\mathcal{I}} \cup D^{\mathcal{I}}$;
- $(\forall R.C)^{\mathcal{I}} = \{x \in \Delta^{\mathcal{I}} : (\forall y \in \Delta^{\mathcal{I}})((x, y) \in R^{\mathcal{I}} \text{ implies } y \in C^{\mathcal{I}})\}$;
- $(\exists R.C)^{\mathcal{I}} = \{x \in \Delta^{\mathcal{I}} : (\exists y \in \Delta^{\mathcal{I}})((x, y) \in R^{\mathcal{I}} \text{ and } y \in C^{\mathcal{I}})\}$.

The interpretation \mathcal{I} satisfies the subsumption $C \sqsubseteq D$ iff $C^{\mathcal{I}} \subseteq D^{\mathcal{I}}$. An interpretation \mathcal{I} is a model of the KB T if it satisfies every subsumption in T . A KB T is said to be consistent or satisfiable if it has a model, whereas a concept expression $C \in \widehat{\mathcal{C}}$ is said to be satisfiable with respect to, or relative to, T if T has a model \mathcal{I} , such that $C^{\mathcal{I}} \neq \emptyset$.

4.2.2 F- \mathcal{ALCI} Basics

In this section, we revisit the basic definitions concerning the syntax and semantics of the modular ontology language F- \mathcal{ALCI} that was introduced in [97]. This language will constitute one of the basic underlying components of the probabilistic counterpart that will be presented in the following sections.

A directed acyclic graph $G = \langle V, E \rangle$ is given, whose vertices represent modules of a federated ontology and whose edges correspond to direct importing relations between the modules. In other words, if $(i, j) \in E$, then module j may import concept names, role names and logical connectives from module i . Note that F- \mathcal{ALCI} is the first modular ontology language that supports contextualization of all logical connectives, rather than just logical negation, as was done in previous proposals [14, 29, 8]. The language of the i -th module in F- \mathcal{ALCI} consists of a set of role expressions $\widehat{\mathcal{R}}_i$ and concept expressions $\widehat{\mathcal{C}}_i$, that are built starting from disjoint collections of concept names \mathcal{C}_i and role names \mathcal{R}_i , for each module $i \in V$. The i -th role expressions are of the form R or R^- , where $R \in \mathcal{R}_j$, $(j, i) \in E$. The i -th concept expressions are built recursively by

$$\top_j, \perp_j, A, \neg_j C, C \sqcap_j D, C \sqcup_j D, \forall_j R.C \text{ and } \exists_j R.C,$$

where $A \in \mathcal{C}_j$, $C, D \in \widehat{\mathcal{C}}_i \cap \widehat{\mathcal{C}}_j$ and $R \in \widehat{\mathcal{R}}_i \cap \widehat{\mathcal{R}}_j$, for $(j, i) \in E$. An i -subsumption in F- \mathcal{ALCI} is a formula of the form $C \sqsubseteq D$, where $C, D \in \widehat{\mathcal{C}}_i$. An ontology $T = \{T_i\}_{i \in V}$ (also known as knowledge base (KB) or TBox) is a V -indexed collection of finite sets T_i of i -subsumptions. The language is provided a formal semantics as follows: An interpretation for T is a pair $\mathcal{I} = \langle \{\mathcal{I}_i\}_{i \in V}, \{r_{ij}\}_{(i,j) \in E} \rangle$, where $\mathcal{I}_i = \langle \Delta^i, \cdot^i \rangle$ is a local interpretation and $r_{ij} \subseteq \Delta^i \times \Delta^j$ is an image-domain relation. The local interpretations $\mathcal{I}_i = \langle \Delta^i, \cdot^i \rangle$ consist of a nonempty local

domain Δ^i and an interpretation function \cdot^i , that assigns to each i -concept name C a set $C^i \subseteq \Delta^i$ and to each i -role name R a set $R^i \subseteq \Delta^i \times \Delta^i$, such that $\top_i^i = \Delta^i$ and $\perp_i^i = \emptyset$. One uses the recursive nature of the concept expressions to extend the functions \cdot^i over all i -role and i -concept expressions as follows:

First, we introduce some notation. For a binary relation $r \subseteq \Delta^i \times \Delta^j$, $X \subseteq \Delta^i$ and $S \subseteq \Delta^i \times \Delta^i$, we set

$$r(X) := \{y \in \Delta^j : (\exists x \in X)((x, y) \in r)\},$$

$$r(S) := \{(z, w) \in \Delta^j \times \Delta^j : (\exists (x, y) \in S)((x, z), (y, w) \in r)\}$$

to denote the images of X and S under the binary relation r .

Let $(R^-)^i = (R^i)^-$, for every $R \in \mathcal{R}_j$, $(j, i) \in E$. Then, for every $A \in \mathcal{C}_j$ and all concept expressions $C, D \in \widehat{\mathcal{C}}_i \cap \widehat{\mathcal{C}}_j$ and role expressions $R \in \widehat{\mathcal{R}}_i \cap \widehat{\mathcal{R}}_j$, with $(j, i) \in E$,

- $\top_j^i = r_{ji}(\Delta^j)$ and $\perp_j^i = \emptyset$;
- $A^i = r_{ji}(A^j)$;
- $(\neg_j C)^i = r_{ji}(\Delta^j \setminus C^j)$;
- $(C \sqcap_j D)^i = r_{ji}(C^j \cap D^j)$;
- $(C \sqcup_j D)^i = r_{ji}(C^j \cup D^j)$;
- $(\forall_j R.C)^i = r_{ji}(\{x \in \Delta^j : (\forall y \in \Delta^j)((x, y) \in R^j \text{ implies } y \in C^j)\})$;
- $(\exists_j R.C)^i = r_{ji}(\{x \in \Delta^j : (\exists y \in C^j)((x, y) \in R^j)\})$.

The interpretation \mathcal{I} **satisfies** the i -subsumption $C \sqsubseteq D$ **as witnessed by** module i iff $C^i \subseteq D^i$. An interpretation \mathcal{I} **satisfies**, or is a **model** of, the KB $T = \{T_i\}_{i \in V}$ if it satisfies every i -subsumption in T_i as witnessed by i , for all $i \in V$. A KB T is said to be **consistent** or **satisfiable** if it has a model \mathcal{I} . On the other hand, an i -concept expression $C \in \widehat{\mathcal{C}}_i$ is said to be **satisfiable as witnessed by i with respect to**, or **relative to**, T if T has a model \mathcal{I} , such that $C^i \neq \emptyset$. Finally, a collection $\mathcal{E} = \{\mathcal{E}_i\}_{i \in V}$, where $\mathcal{E}_i \subseteq \widehat{\mathcal{C}}_i$, is **satisfiable relative to** (or **with respect to**) T if, there exists an interpretation \mathcal{I} (which is a model of T), such that

$\bigcap \{C^i : C \in \mathcal{E}_i\} \neq \emptyset$, for all $i \in V$. When T is empty, we say that \mathcal{E} is **satisfiable** omitting the reference relative to the empty TBox. Note that, because we are assuming that all local domains of every model are nonempty, satisfiability of $\{\mathcal{E}_i\}_{i \in V}$ relative to T is equivalent to the satisfiability of $\{\mathcal{E}_i \cup \{\top_i\}\}_{i \in V}$, relative to T .

4.3 The Probabilistic Extension of F- \mathcal{ALCI}

To introduce the syntax of PF- \mathcal{ALCI} , we define first the concept of conditional constraint. It was given in [61] and forms a cornerstone in the definitions of both P- $\mathcal{SHIF}(\mathbf{D})$ and P- $\mathcal{SHOIN}(\mathbf{D})$ in [64]. To define the semantics of the new probabilistic language involving conditional constraints, the notion of lexicographic entailment, introduced by Lehmann in [57] in the context of default reasoning from conditional knowledge bases, will be employed. This type of reasoning has also been employed in the context of probabilistic default reasoning in [63, 62] and in the definition of the semantics of both P- $\mathcal{SHIF}(\mathbf{D})$ and P- $\mathcal{SHOIN}(\mathbf{D})$ in [64].

4.3.1 Syntax

Roughly speaking, a PF- \mathcal{ALCI} knowledge base or PF- \mathcal{ALCI} ontology is a collection of PTboxes $\{\text{PT}_i\}_{i \in V}$, each of which is an ordinary module of an F- \mathcal{ALCI} knowledge base along with axioms for terminological probabilistic knowledge and default knowledge. An example of an ordinary F- \mathcal{ALCI} -ontology follows.

Example 1: Consider an ontology T with two modules. Module T_1 consists of information regarding the insurance status of the personnel at a given university. It has a concept corresponding to insured personnel and also one corresponding to fully insured and one corresponding to partially insured personnel:

$$\begin{aligned} \text{FullyInsured} &\sqsubseteq \text{Insured} \\ \text{PartiallyInsured} &\sqsubseteq \text{Insured} \\ \text{PartiallyInsured} &\sqsubseteq \neg_1 \text{FullyInsured} \end{aligned}$$

Module T_2 , on the other hand, consists of information about the titles of the personnel, i.e.,

their role in the university structure. It contains concepts for lecturers, faculty, male and female lectures and imports the concept **Insured** from T_1 (equality between two concept expressions stands for subsumption in both directions):

$$\begin{aligned} \text{MaleLecturer} &\sqsubseteq \text{Lecturer} \\ \text{FemaleLecturer} &= \text{Lecturer} \sqcap_2 \neg_2 \text{MaleLecturer} \\ \text{Lecturer} &\sqsubseteq \text{Faculty} \\ \text{Faculty} &\sqsubseteq \text{Insured} \end{aligned}$$

□

Let \mathcal{B}_i be a finite nonempty set of **basic classification i -concepts**, which are i -concept expressions C in $\text{F-}\mathcal{ALCI}$, i.e., $\mathcal{B}_i \subseteq \widehat{\mathcal{C}}_i$. These are the concepts that will be used in conditional constraints to define terminological probabilistic relationships. They will also be used in the semantics of $\text{PF-}\mathcal{ALCI}$ to obtain finite sets of worlds. A **classification i -concept** is defined by recursion starting from basic classification i -concepts as follows:

- Every basic classification i -concept $\phi \in \mathcal{B}_i$ is a classification i -concept.
- If ϕ, ψ are classification i -concepts, then $\neg_i \phi, \phi \sqcap_i \psi, \phi \sqcup_i \psi$ are also classification i -concepts.

The collection of all classification i -concepts is denoted by $\widehat{\mathcal{B}}_i$.

An **i -conditional constraint** is an expression of the form $(\psi|\phi)[l, u]$, where ϕ, ψ are classification i -concepts and $l, u \in [0, 1]$ are reals in the unit interval. This constraint formally expresses the statement that the conditional probability of ψ given ϕ lies between l and u .

As Lukasiewicz observes in [64], the use of classification concepts rather than of only basic classification concepts adds flexibility, reduces the number of worlds that need to be considered in the semantics and brings the framework closer to probabilistic lexicographic entailment in probabilistic default reasoning [63, 62].

Example 2: Assume that all three 1-concept names that we have seen in Example 1, together with another one (that we have not used yet) **HasDental**, with the intended meaning that member employees have dental insurance, are basic classification 1-concepts. The terminolog-

ical probabilistic knowledge “generally, insured personnel are fully insured with probability at least 0.8”, i.e., “typically, a randomly chosen insured employee is fully insured with probability of at least 0.8” can be expressed by the conditional constraint

$$(\text{FullyInsured}|\text{Insured})[0.8, 1].$$

On the other hand, the terminological default knowledge “generally, insured personnel have dental insurance” can be expressed by

$$(\text{HasDental}|\text{Insured})[1, 1]$$

and the default knowledge “generally, partially insured personnel do not have dental insurance” by

$$(\neg_1 \text{HasDental}|\text{PartiallyInsured})[1, 1].$$

This is different from the strict terminological knowledge “all insured employees have dental insurance”, which is expressed by the concept subsumption $\text{Insured} \sqsubseteq \text{HasDental}$. The difference lies in the way these two assertions are handled when used to draw conclusions. More details on this point will come later.

To illustrate our modular approach, we consider also some basic classification 2-concepts and some 2-conditional constraints. Alongside **Faculty** and **Lecturer**, the imported 1-concept name **FullyInsured** and another 2-concept name, that we have not met yet, **DoesResearch**, are basic-classification 2-concepts. Here we have the default knowledge

$$(\text{DoesResearch}|\text{Faculty})[1, 1]$$

$$(\neg_2 \text{DoesResearch}|\text{Lecturer})[1, 1]$$

$$(\text{FullyInsured}|\text{Faculty})[1, 1]$$

and the terminological probabilistic knowledge

$$(\text{FullyInsured}|\text{Lecturer})[0.7, 1].$$

□

A **PF- \mathcal{ALCI} -knowledge base** or **PF- \mathcal{ALCI} -ontology** $PT = \{PT_i\}_{i \in V}$ is a collection of **PTBoxes** $PT_i = \langle T_i, P_i \rangle$, where T_i is the i -**TBox** of an **F- \mathcal{ALCI}** knowledge base $T = \{T_i\}_{i \in V}$ and P_i is a finite set of i -conditional constraints. P_i encodes both probabilistic terminological knowledge and terminological default knowledge. In particular, a specific i -conditional constraint $(\psi|\phi)[l, u]$ has the intended meaning that “generally, if $\phi(a)$ holds, then $\psi(a)$ holds with probability at least l and at most u ”, for every randomly chosen individual a in the domain of discourse.

4.3.2 Semantics

In this section the key concepts of consistency and lexicographic entailment for a **PF- \mathcal{ALCI}** knowledge base will be introduced. The inspiration comes from the work of Lehmann [57] on lexicographic entailment in default reasoning from conditional knowledge bases. Lukasiewicz used this notion to define lexicographic entailment in probabilistic default reasoning in [63, 62] and, more recently, in [64] to obtain lexicographic entailment for his probabilistic description logics. We rely on his latest work to develop the semantics for our framework.

Our goal in reasoning with **PF- \mathcal{ALCI}** is to define new terminological probabilistic knowledge from a given **PF- \mathcal{ALCI}** knowledge base $PT = \{\langle T_i, P_i \rangle\}_{i \in V}$. To perform this reasoning, contextual inconsistencies inside each **PTBox** $PT_i = \langle T_i, P_i \rangle$ have to be resolved. For instance, if the **PTBox** PT_2 includes the probabilistic default statements

$$\begin{aligned} &(\text{DoesResearch}|\text{Faculty})[1, 1] \\ &(\neg_2 \text{DoesResearch}|\text{Lecturer})[1, 1], \end{aligned} \tag{4.1}$$

then an inconsistency is created, given the strict terminological knowledge axiom $\text{Lecturer} \sqsubseteq \text{Faculty}$. Following [64], we use the maximum specificity rule to resolve such inconsistencies. This rule stipulates that more specific information is preferred over less specific one. Since “lecturers do not generally conduct research” is more specific than “faculty do in general conduct research”, the first probability statement in (4.1) will be ignored to resolve this inconsistency.

More generally, the specificity of each conditional constraint in each probabilistic box P_i is analyzed and this analysis leads to establishing a preference relation between all subsets of P_i , which extends to a preference relation between all probabilistic interpretations. This relation is the one that will be used to resolve inconsistencies and draw conclusions whenever possible, i.e., in all cases when the knowledge base is consistent.

4.3.2.1 World Models and Probabilistic Models

Given a collection $\mathcal{E} \subseteq \widehat{\mathcal{C}}_i$, denote by $\neg_i \mathcal{E}$ the set $\neg_i \mathcal{E} = \{\neg_i \phi : \phi \in \mathcal{E}\}$. Let $I = \{I_i\}_{i \in V}$ be a collection of sets of basic classification i -concepts, such that $\{I_i \cup \neg_i(\mathcal{B}_i \setminus I_i)\}_{i \in V}$ is satisfiable¹. I is called a **world relative to $\mathcal{B} = \{\mathcal{B}_i\}_{i \in V}$** . The set of all worlds relative to \mathcal{B} will be denoted by $\mathcal{I}_{\mathcal{B}}$. Since, for all $i \in V$, $|\mathcal{B}_i| < \omega$, we also have that $|\mathcal{I}_{\mathcal{B}}| < \omega$.

Example 3: Consider the knowledge base K that was discussed in the previous examples. We have that

$$\begin{aligned} \mathcal{B}_1 &= \{\text{Insured}, \text{PartiallyInsured}, \text{FullyInsured}, \text{HasDental}\} \\ \mathcal{B}_2 &= \{\text{Faculty}, \text{Lecturer}, \text{FullyInsured}, \text{DoesResearch}\} \end{aligned}$$

Clearly, every $I = \{I_1, I_2\}$, with $I_1 \subseteq \mathcal{B}_1$, $I_2 \subseteq \mathcal{B}_2$, yields a world relative to \mathcal{B} . Thus, in this example, there are $2^4 \cdot 2^4$ worlds relative to \mathcal{B} . \square

Given a world $I = \{I_i\}_{i \in V}$ and an F- \mathcal{ALCI} knowledge base $T = \{T_i\}_{i \in V}$, I **satisfies** T or I is a **model** of T , written $I \models T$, if $\{I_i \cup \neg_i(\mathcal{B}_i \setminus I_i)\}_{i \in V}$ is satisfiable *relative to T* . I **satisfies** a basic classification i -concept $\phi \in \mathcal{B}_i$ or I is a **model** of ϕ , denoted by $I \models_i \phi$, if $\phi \in I_i$. Satisfaction of classification i -concepts by worlds is defined by extending the definition inductively over Boolean connectives in the usual way.

The following proposition is an analog of Proposition 4.8 of [64] and shows that an F- \mathcal{ALCI} knowledge base $T = \{T_i\}_{i \in V}$ is satisfiable iff it has a world model.

¹Recall that this means that there exists an interpretation $\mathcal{I} = \langle \{\mathcal{I}_i\}_{i \in V}, \{r_{ij}\}_{(i,j) \in E} \rangle$, such that $\bigcap \{C^i : C \in I_i \cup \neg_i(\mathcal{B}_i \setminus I_i)\} \neq \emptyset$, for all $i \in V$.

Proposition 47 *Let $\mathcal{B} = \{\mathcal{B}_i\}_{i \in V}$, $\mathcal{B}_i \neq \emptyset$, be a family of finite sets of basic classification i -concepts and $T = \{T_i\}_{i \in V}$ an F- \mathcal{ALCT} knowledge base. T has a model $\mathcal{I} = \langle \{\mathcal{I}_i\}_{i \in V}, \{r_{ij}\}_{(i,j) \in E} \rangle$, with $\mathcal{I}_i = \langle \Delta^i, \cdot^i \rangle$, $i \in V$, iff T has a world model $I = \{I_i\}_{i \in V}$ relative to \mathcal{B} .*

Proof:

Suppose, first, that $\mathcal{I} = \langle \{\mathcal{I}_i\}_{i \in V}, \{r_{ij}\}_{(i,j) \in E} \rangle$ is a model of $T = \{T_i\}_{i \in V}$, with $\mathcal{I}_i = \langle \Delta^i, \cdot^i \rangle$, $i \in V$. Recall that we are assuming that $\Delta^i \neq \emptyset$, for all $i \in V$. Let, for each $i \in V$, $a_i \in \Delta^i$. Define $I_i = \{\phi \in \mathcal{B}_i : a_i \in \phi^i\}$, $i \in V$. Then $I = \{I_i\}_{i \in V}$ is a world relative to \mathcal{B} that is also a model of T . If, conversely, T has a world model $I = \{I_i\}_{i \in V} \in \mathcal{I}_{\mathcal{B}}$, then $\{I_i \cup \neg_i(\mathcal{B}_i \setminus I_i)\}_{i \in V}$ is satisfiable relative to T , whence T_i is a fortiori satisfiable. \square

A **probabilistic interpretation** Pr is a probability function on the set of all worlds $\mathcal{I}_{\mathcal{B}}$ over the set \mathcal{B} of basic classification concepts, i.e., a mapping $\text{Pr} : \mathcal{I}_{\mathcal{B}} \rightarrow [0, 1]$, such that $\sum_{I \in \mathcal{I}_{\mathcal{B}}} \text{Pr}(I) = 1$. Pr **satisfies** an F- \mathcal{ALCT} knowledge base $T = \{T_i\}_{i \in V}$ or Pr is a **model** of T , denoted $\text{Pr} \models T$, if, for all $I \in \mathcal{I}_{\mathcal{B}}$, such that $\text{Pr}(I) > 0$, $I \models T$. As far as satisfaction of conditional constraints goes, we set it up as follows: The **probability** of a classification i -concept ϕ in a probabilistic interpretation Pr , denoted $\text{Pr}_i(\phi)$, is defined by

$$\text{Pr}_i(\phi) = \sum \{\text{Pr}(I) : I \models_i \phi\}.$$

Furthermore, for all classification i -concepts ϕ and ψ , such that $\text{Pr}_i(\phi) > 0$, we set

$$\text{Pr}_i(\psi|\phi) = \frac{\text{Pr}_i(\phi \sqcap_i \psi)}{\text{Pr}_i(\phi)}.$$

Pr **satisfies** an i -conditional constraint $(\psi|\phi)[l, u]$ or Pr is a **model** of $(\psi|\phi)[l, u]$, denoted $\text{Pr} \models_i (\psi|\phi)[l, u]$, if $\text{Pr}_i(\phi) = 0$ or $\text{Pr}_i(\psi|\phi) \in [l, u]$. Pr **satisfies** a set of i -conditional constraints \mathcal{F}_i or Pr is a **model** of \mathcal{F}_i , written $\text{Pr} \models_i \mathcal{F}_i$, if $\text{Pr} \models_i F$, for all $F \in \mathcal{F}_i$. Finally, if $\mathcal{F} = \{\mathcal{F}_i\}_{i \in V}$ is a collection of sets of i -conditional constraints for $i \in V$, we write $\text{Pr} \models \mathcal{F}$ to signify that $\text{Pr} \models_i \mathcal{F}_i$, for all $i \in V$.

In Proposition 48, an analog of Proposition 4.9 of [64] in the federated setting, it is shown that an F- \mathcal{ALCT} knowledge base $T = \{T_i\}_{i \in V}$ is satisfiable if and only if it has a probabilistic

model.

Proposition 48 *Let $\mathcal{B} = \{\mathcal{B}_i\}_{i \in V}$ be a collection of nonempty sets of basic classification concepts and $T = \{T_i\}_{i \in V}$ an F- \mathcal{ALCI} knowledge base. T has a model $\mathcal{I} = \langle \{\mathcal{I}_i\}_{i \in V}, \{r_{ij}\}_{(i,j) \in E} \rangle$, with $\mathcal{I}_i = \langle \Delta^i, \cdot^i \rangle, i \in V$, if and only if it has a probabilistic model Pr on $\mathcal{I}_{\mathcal{B}}$.*

Proof:

Suppose that T has a model $\mathcal{I} = \langle \{\mathcal{I}_i\}_{i \in V}, \{r_{ij}\}_{(i,j) \in E} \rangle$. By Proposition 47, T has a world model $I = \{I_i\}_{i \in V}$. Define $\text{Pr}(I) = 1$ and $\text{Pr}(I') = 0$, for all $I' \in \mathcal{I}_{\mathcal{B}}$, with $I' \neq I$. Then Pr is a probabilistic model of T on $\mathcal{I}_{\mathcal{B}}$. Suppose, conversely, that T has a probabilistic model Pr on $\mathcal{I}_{\mathcal{B}}$. Then, there exists $I \in \mathcal{I}_{\mathcal{B}}$, such that $\text{Pr}(I) > 0$. Since $\text{Pr} \models T$, this implies that $I \models T$. Hence, again by Proposition 47, T has a model $\mathcal{I} = \langle \{\mathcal{I}_i\}_{i \in V}, \{r_{ij}\}_{(i,j) \in E} \rangle$. \square

4.3.2.2 z -Partitions and Consistency

A probabilistic interpretation Pr **verifies** an i -conditional constraint $(\psi|\phi)[l, u]$ if $\text{Pr}_i(\phi) = 1$ and $\text{Pr}_i(\psi) \in [l, u]$ (see also [63, 62]). On the other hand, Pr **falsifies** $(\psi|\phi)[l, u]$ if $\text{Pr}_i(\phi) = 1$ and $\text{Pr} \not\models_i (\psi|\phi)[l, u]$. A collection $\mathcal{F} = \{\mathcal{F}_i\}_{i \in V}$ of sets of conditional constraints **tolerates** an i -conditional constraint $(\psi|\phi)[l, u]$ **under** an F- \mathcal{ALCI} knowledge base $T = \{T_i\}_{i \in V}$, or $(\psi|\phi)[l, u]$ **is tolerated under T by \mathcal{F}** , if $T \cup \mathcal{F}$ has a model that verifies $(\psi|\phi)[l, u]$.

Example 4: We illustrate, using our previous examples, ways in which tolerance of an i -conditional constraint by a collection of sets of conditional constraints may fail. We consider again the F- \mathcal{ALCI} TBox $T = \{T_1, T_2\}$ of Example 1 and we set

$$\begin{aligned} \mathcal{F}_1 &= \{(\text{FullyInsured}|\text{Insured})[0.8, 1], (\text{HasDental}|\text{Insured})[1, 1]\} \\ \mathcal{F}_2 &= \{(\text{DoesResearch}|\text{Faculty})[1, 1], (\text{FullyInsured}|\text{Faculty})[1, 1]\} \end{aligned}$$

and $\mathcal{F} = \{\mathcal{F}_1, \mathcal{F}_2\}$. Assume, for the sake of obtaining a contradiction that

$$F_1 = (\neg_1 \text{HasDental}|\text{PartiallyInsured})[1, 1]$$

is tolerated under T by \mathcal{F} . Let Pr be a probabilistic model of $T \cup \mathcal{F}$ that verifies F_1 . Then, $\text{Pr}_1(\text{PartiallyInsured}) = 1$ and $\text{Pr}_1(\neg_1 \text{HasDental}) = 1$. But, since

$$\text{PartiallyInsured} \sqsubseteq \text{Insured} \in T_1,$$

$\text{Pr}_1(\text{PartiallyInsured}) = 1$ implies that $\text{Pr}_1(\text{Insured}) = 1$, whence, since

$$(\text{HasDental}|\text{Insured})[1, 1] \in \mathcal{F}_1$$

and Pr is a model of \mathcal{F} , we get that $\text{Pr}_1(\text{HasDental}) = 1$. This clearly contradicts the statement $\text{Pr}_1(\neg_1 \text{HasDental}) = 1$. Therefore, F_1 is not tolerated under T by \mathcal{F} .

Similarly, it is easily seen that $F_2 = (\neg_2 \text{DoesResearch}|\text{Lecturer})[1, 1]$ is not tolerated under T by \mathcal{F} because, if one assumes that a person is a lecturer, then they are clearly faculty, who are assumed, in general, to perform research and this would contradict the information that lecturers, in general, do not do research. If, on the other hand, our sets of conditional constraints included the more specific information represented by F_1 and F_2 as opposed to the facts that insured people, in general, have dental insurance and faculty, in general, do research, respectively, then the more general information would be tolerated by the more specific pieces of information. \square

Concerning tolerance, it is not difficult to see that the following proposition, relating tolerance with the existence of a probabilistic model, which is due essentially to Lukasiewicz [64], holds:

Proposition 49 *An i -conditional constraint $(\psi|\phi)[l, u]$ is tolerated under an F -ALCI knowledge base $T = \{T_i\}_{i \in V}$ by a collection $\mathcal{F} = \{\mathcal{F}_i\}_{i \in V}$ of sets of conditional constraints iff there exists a probabilistic model Pr of $T \cup \mathcal{F}'$, where $\mathcal{F}' = \{\mathcal{F}'_i\}_{i \in V}$, with*

$$\mathcal{F}'_j = \begin{cases} \mathcal{F}_i \cup \{(\psi|\phi)[l, u], (\phi|\top_i)[1, 1]\}, & \text{if } j = i \\ \mathcal{F}_j, & \text{otherwise} \end{cases}$$

Proof:

Suppose, first, that $(\psi|\phi)[l, u]$ is tolerated under T by \mathcal{F} . Thus, by definition, $T \cup \mathcal{F}$ has a model Pr verifying $(\psi|\phi)[l, u]$. Hence $\text{Pr}_i(\phi) = 1$ and $\text{Pr}_i(\psi) \in [l, u]$. But then $\text{Pr}_i(\phi|\top_i) = \frac{\text{Pr}_i(\phi \sqcap_i \top_i)}{\text{Pr}_i(\top_i)} = \text{Pr}_i(\phi) = 1$ and also $\text{Pr}_i(\psi|\phi) = \frac{\text{Pr}_i(\phi \sqcap_i \psi)}{\text{Pr}_i(\phi)} = \text{Pr}_i(\phi \sqcap_i \psi) = \text{Pr}_i(\psi) \in [l, u]$. Therefore, Pr is a probabilistic model of $T \cup \mathcal{F}'$. For the converse we follow the reverse steps. \square

A PF- \mathcal{ALCC} knowledge base $\text{PT} = \{\text{PT}_i\}_{i \in V}$, with $\text{PT}_i = \langle T_i, P_i \rangle$, is **consistent** if

- (i) $T = \{T_i\}_{i \in V}$ is satisfiable;
- (ii) There exist $k_i \geq 0, i \in V$, and ordered partitions $(P_0^i, \dots, P_{k_i}^i)$ of P_i , such that each P_j^i , $j = 0, \dots, k_i$, is the set of all $F \in P_i \setminus (P_0^i \cup \dots \cup P_{j-1}^i)$ that are tolerated under T by $P \setminus (P_0 \cup \dots \cup P_{j-1})$, where

$$P \setminus (P_0 \cup \dots \cup P_{j-1}) = \{P_i \setminus (P_0^i \cup \dots \cup P_{j-1}^i)\}_{i \in V}.$$

(In case, for some $i \in V$, $j > k_i$, we set $P_j^i = \emptyset$.)

The ordered partitions $(P_0^i, \dots, P_{k_i}^i)$ are unique if they exist. Taking after [64], the term **z -partition** of $\text{PT} = \{\text{PT}_i\}_{i \in V}$ will be used to refer to these partitions. Intuitively speaking, the z -partition enables one to resolve contextual inconsistencies by selecting more specific conditional constraints over less specific ones, as was demonstrated in Example 4.

Example 5: Consider again the TBox $T = \{T_1, T_2\}$ of Example 1 and the PBox $P = \{P_1, P_2\}$ discussed in Example 2. It is not difficult to see that T is satisfiable and that there exists a unique z -partition $P_1 = (P_0^1, P_1^1), P_2 = (P_0^2, P_1^2)$, given by

$$\begin{aligned} P_0^1 &= \{(\psi|\phi)[l, u] \in P_1 : \phi = \text{Insured}\} \\ P_1^1 &= \{(\neg_1 \text{HasDental} | \text{PartiallyInsured})[1, 1]\} \end{aligned}$$

and

$$\begin{aligned} P_0^2 &= P_2 \setminus \{(\neg_2 \text{DoesResearch} | \text{Lecturer})[1, 1]\} \\ P_1^2 &= \{(\neg_2 \text{DoesResearch} | \text{Lecturer})[1, 1]\} \end{aligned}$$

Thus, the knowledge base $\text{PT} = \{\text{PT}_1 \text{PT}_2\}$ is consistent. Note that, in the construction of the z -partition of P_2 , the 2-conditional constraint $(\text{FullyInsured} | \text{Lecturer})[0.7, 1]$ is in the first block; it is tolerated under T by P , even though P_2 contains the “more general” piece of information $(\text{FullyInsured} | \text{Faculty})[1, 1]$. \square

4.3.2.3 Lexicographic Entailment

Let $\text{PT} = \{\langle T_i, P_i \rangle\}_{i \in V}$ be a consistent PF- \mathcal{ALCI} knowledge base, with z -partition

$$\{(P_0^i, \dots, P_{k_i}^i) : i \in V\}.$$

We follow [64] in first defining a lexicographic preference relation on probabilistic interpretations and, then, a lexicographic entailment for sets of conditional constraints under PTBoxes.

Given two probabilistic interpretations Pr and Pr' , Pr is said to be **lexicographically preferable** or **lex-preferable** to Pr' if, for all $i \in V$, there exists $j_i \in \{0, 1, \dots, k_i\}$, such that

- $|\{F \in P_{j_i}^i : \text{Pr} \models_i F\}| > |\{F \in P_{j_i}^i : \text{Pr}' \models_i F\}|$ and
- $|\{F \in P_l^i : \text{Pr} \models_i F\}| = |\{F \in P_l^i : \text{Pr}' \models_i F\}|$, for all $j_i < l \leq k_i$.

The lex-preference relation implements the idea of preferring more specific sets of conditional constraints to less specific ones. This is used to resolve contextual inconsistencies when drawing conclusions.

A probabilistic interpretation Pr , satisfying an F- \mathcal{ALCI} knowledge base $T = \{T_i\}_{i \in V}$ and a collection $\mathcal{F} = \{\mathcal{F}_i\}_{i \in V}$ of sets of conditional constraints is a **lexicographically minimal** or **lex-minimal model** of $T \cup \mathcal{F}$ if no model of $T \cup \mathcal{F}$ is lex-preferable to Pr .

An i -conditional constraint $(\psi | \phi)[l, u]$ is a **lexicographic-consequence** or **lex-consequence of** a collection of sets of conditional constraints $\mathcal{F} = \{\mathcal{F}_i\}_{i \in V}$ **under** a PTBox PT , written $\mathcal{F} \Vdash_i^{\text{lex}} (\psi | \phi)[l, u]$ **under** PT if $\text{Pr}_i(\psi) \in [l, u]$, for every lex-minimal model Pr of $T \cup \mathcal{F}'$,

where $\mathcal{F}' = \{\mathcal{F}'_i\}_{i \in V}$, with

$$\mathcal{F}'_j = \begin{cases} \mathcal{F}_i \cup \{(\phi|\top_i)[1, 1]\}, & \text{if } j = i \\ \mathcal{F}_j, & \text{otherwise.} \end{cases} \quad (4.2)$$

An i -conditional constraint $(\psi|\phi)[l, u]$ is a **tight lexicographic-consequence** or **tight lex-consequence of $\mathcal{F} = \{\mathcal{F}_i\}_{i \in V}$ under PT**, written $\mathcal{F} \Vdash_i^{\text{tlex}} (\psi|\phi)[l, u]$ **under PT** if l and u are, respectively, the infimum and supremum of $\text{Pr}_i(\psi)$ subject to all lex-minimal models Pr of $T \cup \mathcal{F}'$, with \mathcal{F}' as in Equation (4.2).

Note that $[l, u] = [1, 0]$ when no such models exist. Moreover, we stipulate that $\mathcal{F} \Vdash_i^{\text{lex}} (\psi|\phi)[l, u]$ and $\mathcal{F} \Vdash_i^{\text{tlex}} (\psi|\phi)[1, 0]$ under PT, for all \mathcal{F} and all $(\psi|\phi)[l, u]$ in case PT is an inconsistent PTBox.

Example 6: Consider again the knowledge base PT with TBox introduced in Example 1 and PBox introduced in Example 2. The analysis performed in Examples 4 and 5, together with the definition of lex-preference and lex-consequence, show that, for any three probabilistic interpretations Pr, Pr' and Pr'' , such that

- (a) $\text{Pr} \models P$;
- (b) $\text{Pr}' \models P \setminus \{(\text{HasDental}|\text{Insured})[1, 1], (\text{DoesResearch}|\text{Faculty})[1, 1]\}$;
- (c) $\text{Pr}'' \models P \setminus \{(\neg_1 \text{HasDental}|\text{PartiallyInsured})[1, 1],$
 $(\neg_2 \text{DoesResearch}|\text{Lecturer})[1, 1]\}$

Pr is lex-preferable to both Pr' and Pr'' and Pr' is lex-preferable to Pr'' . Thus, according to lex-preference, any probabilistic interpretation Pr , as above, will be preferred to Pr' , resulting in the more specific 1-conditional constraint $(\neg_1 \text{HasDental}|\text{PartiallyInsured})[1, 1] \in P_1^1$ to be preferred in reasoning over the more general 1-conditional constraint

$$(\text{HasDental}|\text{Insured})[1, 1] \in P_0^1.$$

Similar comments apply in comparing the more specific 2-conditional constraint

$$(\neg_2 \text{DoesResearch} | \text{Lecturer})[1, 1] \in P_1^2$$

with the more general 2-conditional constraint $(\text{DoesResearch} | \text{Faculty}) [1, 1] \in P_0^2$. These choices will be used when deriving lex-consequences to resolve conflicts involving default information. \square

An i -conditional constraint F is a **lex-consequence** of a PTBox PT, denoted $\text{PT} \Vdash_i^{\text{lex}} F$, if $\emptyset \Vdash_i^{\text{lex}} F$ under PT. F is a **tight lex-consequence** of PT, denoted $\text{PT} \Vdash_i^{\text{tlex}} F$, if $\emptyset \Vdash_i^{\text{tlex}} F$ under PT.

An analog of Theorem 4.18 of [64] provides a characterization of lexicographic entailment for a set of conditional constraints under a PTBox in terms of satisfiability and logical entailment of conditional constraints under an F- \mathcal{ALCI} knowledge base. We introduce some additional definitions to prepare the groundwork for formulating this analog.

Given an F- \mathcal{ALCI} knowledge base $T = \{T_i\}_{i \in V}$ and a V -indexed collection $\mathcal{F} = \{\mathcal{F}_i\}_{i \in V}$ of sets of conditional constraints, $T \cup \mathcal{F}$ is **satisfiable** if there exists a model of $T \cup \mathcal{F}$. An i -conditional constraint $(\psi|\phi)[l, u]$ is a **logical consequence** of $T \cup \mathcal{F}$, denoted $T \cup \mathcal{F} \models_i (\psi|\phi)[l, u]$, if each model of $T \cup \mathcal{F}$ is also a model of $(\psi|\phi)[l, u]$. $(\psi|\phi)[l, u]$ is a **tight logical consequence** of $T \cup \mathcal{F}$, denoted $T \cup \mathcal{F} \models_i^t (\psi|\phi)[l, u]$, if l and u are, respectively, the infimum and supremum of $\text{Pr}_i(\psi|\phi)$ subject to all models Pr of $T \cup \mathcal{F}$, with $\text{Pr}_i(\phi) > 0$.

Let $P = \{P_i\}_{i \in V}$ be the PBox of a PF- \mathcal{ALCI} knowledge base $\text{PT} = \{\text{PT}_i\}_{i \in V}$. An indexed subfamily $Q = \{Q_i\}_{i \in V}$, with $Q_i \subseteq P_i$, $i \in V$, denoted $Q \leq P$, is **lexicographically-preferable** or **lex-preferable** to $Q' = \{Q'_i\}_{i \in V}$, with $Q' \leq P$, if, for all $i \in V$, there exists $j_i \in \{0, 1, \dots, k_i\}$, such that

- $|Q_i \cap P_{j_i}^i| > |Q'_i \cap P_{j_i}^i|$ and
- $|Q_i \cap P_l^i| = |Q'_i \cap P_l^i|$, for all $j_i < l \leq k_i$,

where $\{(P_0^i, \dots, P_{k_i}^i) : i \in V\}$ is the z -partition of PT. Q is **lexicographically-minimal** or **lex-minimal** in $S = \{S^k\}_{k \in I}$, $S^k = \{S_i^k\}_{i \in V}$, $S^k \leq P$, if $Q \in S$ and no $Q' \in S$ is lex-preferable

to Q .

Theorem 50 *Let $\text{PT} = \{\langle T_i, P_i \rangle\}_{i \in V}$ be a consistent PTBox , $\mathcal{F} = \{\mathcal{F}_i\}_{i \in V}$ a family of sets of conditional constraints and ϕ, ψ two i -classification concepts. Consider the collection \mathcal{Q} of all lex-minimal elements in the set of all $Q \leq P$, such that $T \cup Q \cup \mathcal{F}'$ is satisfiable, where $\mathcal{F}' = \{\mathcal{F}'_i\}_{i \in V}$ is given by Equation (4.2).*

1. *If $\mathcal{Q} = \emptyset$, then $\mathcal{F} \Vdash_i^{\text{tlex}} (\psi|\phi)[1, 0]$ under PT .*
2. *If $\mathcal{Q} \neq \emptyset$, then $\mathcal{F} \Vdash_i^{\text{tlex}} (\psi|\phi)[l, u]$ under PT , where $l = \min l'$ and $u = \max u'$, subject to $T \cup Q \cup \mathcal{F}' \models_i^t (\psi|\top_i)[l', u']$ and $Q \in \mathcal{Q}$.*

Proof:

The proof is very similar to the proof of Theorem 4.18 of [64] (see page 878).

1. If $\mathcal{Q} = \emptyset$, then $T \cup Q \cup \mathcal{F}'$ is not satisfiable, for any $Q \leq P$. Thus, $T \cup \mathcal{F}'$ is not satisfiable. This shows that $\mathcal{F} \Vdash_i^{\text{tlex}} (\psi|\phi)[1, 0]$ under PT .
2. Let $\mathcal{Q} \neq \emptyset$. Then Pr is a lex-minimal model of $T \cup \mathcal{F}'$ iff (i) Pr is a model of $T \cup \mathcal{F}'$ and (ii) $\{\{F_i \in P_i : \text{Pr} \models_i F_i\}\}_{i \in V}$ is a lex minimal element in the set of all $Q \leq P$, such that $T \cup Q \cup \mathcal{F}'$ is satisfiable. This shows that Pr is a model of $T \cup Q \cup \mathcal{F}'$, for some $Q \in \mathcal{Q}$. \square

4.4 A Modified P- \mathcal{ALCI}

In the remainder of the paper, we formulate some inference problems concerning PF- \mathcal{ALCI} knowledge bases. Our goal is to reduce these problems to corresponding problems for non-federated probabilistic knowledge bases and, then, use already known procedures from [64] (or slightly modified versions) in order to solve them. Apart from the obvious algorithmic advantage, we also get the side-benefit of being able to pinpoint the algorithmic complexity of the federated problems, based on the complexities of the unimodule versions. The unfortunate fact is that the probabilistic description logic that results by restricting the logic P- $\mathcal{SHIF}(\mathbf{D})$, as presented in [64], by adopting as its underlying description logic \mathcal{ALCI} and by disregarding

its assertional part, does not serve exactly our goal. Intuitively, this happens because the world models in the semantics of $P\text{-}\mathcal{SHIF}(\mathbf{D})$ are based in a single set of basic classification concepts, whereas, our reduction will necessitate the existence of multiple sets of basic classification concepts. For this reason, in this section we present a single module probabilistic language $P\text{-}\mathcal{ALCI}$ by slightly generalizing the semantics of the probabilistic terminological and terminological default knowledge given in [64]. This modified version will be appropriate for accommodating the sound and complete reductions of the inference problems for $PF\text{-}\mathcal{ALCI}$. Furthermore, we introduce the main problems that we will consider in $P\text{-}\mathcal{ALCI}$. They are the same as those of Lukasiewicz [64], but refer to the modified semantics. Finally, we argue that these problems may be solved with algorithms virtually identical to the ones provided by Lukasiewicz and, as a result, maintain the same computational complexities.

4.4.1 Syntax and Semantics

Since in the federated language $PF\text{-}\mathcal{ALCI}$ we deal only with PTBoxes, we restrict our attention in this section to $P\text{-}\mathcal{ALCI}$ knowledge bases with only PTBoxes of the form $PT = \langle T, P \rangle$. We assume that among the \mathcal{ALCI} concept names, there are $|V|$ names $\top_i, i \in V$, whose extensions, intuitively, will represent the parts of the local domains of a $PF\text{-}\mathcal{ALCI}$ interpretation corresponding to the various modules, when combined into a single large domain. The TBox T is an ordinary \mathcal{ALCI} TBox and the semantics concerning the $P\text{-}\mathcal{ALCI}$ roles, concepts and TBox axioms is exactly the ordinary semantics of \mathcal{ALCI} , with the only exception that an \mathcal{ALCI} model $\mathcal{I} = \langle \Delta^{\mathcal{I}}, \cdot^{\mathcal{I}} \rangle$ must satisfy $\top_i^{\mathcal{I}} \neq \emptyset$, for all $i \in V$. To construct the PBox, fix a collection $\{\mathcal{C}_i\}_{i \in V}$ of nonempty sets of **basic classification concepts** or **basic c -concepts**, which are concepts in \mathcal{ALCI} , such that $C \sqsubseteq \top_i \in T$, for every $C \in \mathcal{C}_i, i \in V$. These will be the sets of relevant concepts for defining probabilistic relationships. This is the main deviation from the languages in [64], which have one set of basic classification concepts. We adopt this modification, as explained previously, because we would like to use $P\text{-}\mathcal{ALCI}$ to simulate $PF\text{-}\mathcal{ALCI}$, which employs multiple sets of basic classification concepts each of which is used to classify probabilistic relationships between concepts appearing in a corresponding module of a feder-

ated ontology. Accordingly, we also obtain $|V|$ sets of **classification concepts** or *c-concepts*. These are defined recursively by taking negations, conjunctions and disjunctions starting from the corresponding set of basic *c-concepts*. We also construct conditional constraints of $|V|$ types, each using corresponding *c-concepts*. Thus, a **conditional constraint of type i** is one of the form $(\psi|\phi)[l, u]$, where ϕ, ψ are *c-concepts* from the i -th set and $l, u \in [0, 1]$. We denote by P a finite set of conditional constraints (possibly of many types), called a **PBox**. A **PTBox** $PT = \langle T, P \rangle$ consists of an \mathcal{ALCI} TBox T together with a PBox P . For our purposes a **probabilistic knowledge base** and a PTBox coincide, since we do not consider individuals and, as a result we do not consider either classical ABoxes or PBoxes in the sense of [64].

To accommodate the multiplicity of the sets of basic classification concepts, the notion of a world in the semantics must be modified as compared to the standard one of [64]. Namely, a **world** is a collection of sets $I = \{I_i\}_{i \in V}$, with $I_i \subseteq \mathcal{C}_i$, such that the concept $\bigcap_{C \in I_i} C \sqcap \bigcap_{C \in \mathcal{C}_i \setminus I_i} \neg C$ is satisfiable, for all $i \in V$. The set of all worlds is denoted by \mathcal{I}_C . A world I is said to **satisfy** an \mathcal{ALCI} TBox T or to be a **model** of T , written $I \models T$, if there exists a model $\mathcal{I} = \langle \Delta^{\mathcal{I}}, \cdot^{\mathcal{I}} \rangle$ of T , such that $\bigcap_{C \in I_i} C^{\mathcal{I}} \cap \bigcap_{C \in \mathcal{C}_i \setminus I_i} \Delta^{\mathcal{I}} \setminus C^{\mathcal{I}} \neq \emptyset$, for all $i \in V$. A world $I = \{I_i\}_{i \in V}$ **satisfies** a basic classification concept $\phi \in \mathcal{C}_i$ if $\phi \in I_i$. This notion can be extended in the usual way over classification concepts of type i .

It can be shown, now, using the modified definitions that were introduced above that an \mathcal{ALCI} TBox T has a model $\mathcal{I} = \langle \Delta^{\mathcal{I}}, \cdot^{\mathcal{I}} \rangle$ iff it has a world model $I = \{I_i\}_{i \in V}$.

A **probabilistic interpretation** Pr is a probability function on \mathcal{I}_C , as in the ordinary case [64]. It **satisfies** an \mathcal{ALCI} TBox T or is a **model** of T if $I \models T$, for all I , such that $\text{Pr}(I) > 0$. Moreover, for a classification concept ϕ of type i , we have $\text{Pr}(\phi) = \sum \{\text{Pr}(I) : I \models \phi\}$ and, for two classification concepts ϕ, ψ of type i , such that $\text{Pr}(\phi) > 0$, we define $\text{Pr}(\psi|\phi) = \frac{\text{Pr}(\phi \sqcap \psi)}{\text{Pr}(\phi)}$. Then Pr **satisfies** or is a **model** of a conditional constraint $(\psi|\phi)[l, u]$ of type i , written $\text{Pr} \models (\psi|\phi)[l, u]$, if $\text{Pr}(\phi) = 0$ or $\text{Pr}(\psi|\phi) \in [l, u]$. This notion extends to satisfiability of a set of conditional constraints, possibly consisting of constraints of more than one types.

A probabilistic interpretation is said to **verify** a conditional constraint $(\psi|\phi)[l, u]$ if $\text{Pr}(\phi) = 1$ and $\text{Pr}(\psi) \in [l, u]$. A set \mathcal{F} of conditional constraints (possibly of various types) is said to

tolerate a conditional constraint $(\psi|\phi)[l, u]$ **under** an \mathcal{ALCI} TBox T if $T \cup \mathcal{F}$ has a model that verifies $(\psi|\phi)[l, u]$. These concepts help in defining the notion of consistency for a P- \mathcal{ALCI} PTBox, which is a slightly modified version of the one given in Section 4.2.3 of [64]. A PTBox $PT = \langle T, P \rangle$ is **consistent** if T is satisfiable (i.e., has a model; recall that the $\top_i, i \in V$, must have nonempty extensions in the model) and there exists an ordered partition (P_0, \dots, P_k) of P , such that each P_i is the set of all $F \in P \setminus \bigcup_{j=0}^{i-1} P_j$ that are tolerated by $P \setminus \bigcup_{j=0}^{i-1} P_j$ under T . The ordered partition (P_0, \dots, P_k) is unique, if it exists, and, following [64], we call it the **z-partition** of PT and define $P_j^i = \{F \in P_j : F \text{ is of type } i\}$, for all $j = 1, \dots, k, i \in V$.

To define lexicographic entailment, we fix a consistent P- \mathcal{ALCI} PTBox $PT = \langle T, P \rangle$. Thus, there exists a z -partition (P_0, \dots, P_k) of PT . A probabilistic interpretation Pr is said to be **lexicographically preferable** or **lex-preferable** to a probabilistic interpretation Pr' if, for every $i \in V$, there exists $j_i, 0 \leq j_i \leq k$, such that $|\{F \in P_{j_i}^i : \text{Pr} \models F\}| > |\{F \in P_{j_i}^i : \text{Pr}' \models F\}|$ and $|\{F \in P_l^i : \text{Pr} \models F\}| = |\{F \in P_l^i : \text{Pr}' \models F\}|$, for all $j_i < l \leq k$. A probabilistic interpretation Pr , that satisfies T and a set \mathcal{F} of conditional constraints (of possibly various types) is a **lexicographically minimal** or **lex-minimal model** of $T \cup \mathcal{F}$ if no model of $T \cup \mathcal{F}$ is lex-preferable to Pr . A conditional constraint $(\psi|\phi)[l, u]$ of type i is a **lexicographic consequence** or **lex-consequence** of a set \mathcal{F} of conditional constraints under a PTBox PT , denoted $\mathcal{F} \models^{\text{lex}} (\psi|\phi)[l, u]$ **under** PT , if $\text{Pr}(\psi) \in [l, u]$ for every lex-minimal model Pr of $T \cup \mathcal{F} \cup \{(\phi|\top)[1, 1]\}$. The conditional constraint $(\psi|\phi)[l, u]$ of type i is a **tight lexicographic consequence** or **tight lex-consequence** of a set \mathcal{F} of conditional constraints under a PTBox PT , denoted $\mathcal{F} \models_{\text{tight}}^{\text{lex}} (\psi|\phi)[l, u]$ **under** PT , if l and u are, respectively, the infimum and supremum of $\text{Pr}(\psi)$ subject to all lex-minimal models Pr of $T \cup \mathcal{F} \cup \{(\phi|\top)[1, 1]\}$. For inconsistent PTBoxes PT , we define $\mathcal{F} \models^{\text{lex}} (\psi|\phi)[l, u]$ and $\mathcal{F} \models_{\text{tight}}^{\text{lex}} (\psi|\phi)[1, 0]$ under PT , for all sets of conditional constraints \mathcal{F} and all conditional constraints $(\psi|\phi)[l, u]$. Finally, a conditional constraint F is a **lex-consequence** of a PTBox PT , denoted $PT \models^{\text{lex}} F$, if $\emptyset \models^{\text{lex}} F$ under PT and F is a **tight lex-consequence** of PT , denoted $PT \models_{\text{tight}}^{\text{lex}} F$, if $\emptyset \models_{\text{tight}}^{\text{lex}} F$ under PT .

Along the lines of the characterization of lexicographic entailment for a set of conditional

constraints under a PTBox in terms of satisfiability and logical entailment for a set of conditional constraints under a classical knowledge base that was provided in [64] (Theorem 4.18), one may give a characterization for the modified language P- \mathcal{ALCI} , presented in this section. Given an \mathcal{ALCI} TBox T and a set of conditional constraints \mathcal{F} of possibly various types, $T \cup \mathcal{F}$ is **satisfiable** if a model of $T \cup \mathcal{F}$ exists. Again, such a model in the present context is assumed to assign nonempty extensions to all \top_i , $i \in V$. A conditional constraint $(\psi|\phi)[l, u]$ of type i is a **logical consequence** of $T \cup \mathcal{F}$, denoted $T \cup \mathcal{F} \models (\psi|\phi)[l, u]$, if each model of $T \cup \mathcal{F}$ is also a model of $(\psi|\phi)[l, u]$. The conditional constraint $(\psi|\phi)[l, u]$ is a **tight logical consequence** of $T \cup \mathcal{F}$, denoted $T \cup \mathcal{F} \models_{\text{tight}} (\psi|\phi)[l, u]$, if l and u are, respectively, the infimum and the supremum of $\text{Pr}(\psi|\phi)$ subject to all models Pr of $T \cup \mathcal{F}$, with $\text{Pr}(\phi) > 0$. Let $\text{PT} = \langle T, P \rangle$ be a consistent P- \mathcal{ALCI} PTBox, where (P_0, \dots, P_k) denotes the z -partition of PT and $P_j^i, i \in V, j = 1, \dots, k$ are as before. A subset $Q \subseteq P$ is **lexicographically preferable** or **lex-preferable** to $Q' \subseteq P$ if, for every $i \in V$, there exists a $j_i \in \{0, \dots, k\}$, such that $|Q \cap P_{j_i}^i| > |Q' \cap P_{j_i}^i|$ and $|Q \cap P_l^i| = |Q' \cap P_l^i|$, for all $j_i < l \leq k$. Q is **lexicographically minimal** or **lex-minimal** in $\mathcal{S} \subseteq \mathcal{P}(P)$ if $Q \in \mathcal{S}$ and no $Q' \in \mathcal{S}$ is lex-preferable to Q . Analogously to Theorem 4.18 of [64], we obtain for our modified language P- \mathcal{ALCI} that for a consistent PTBox $\text{PT} = \langle T, P \rangle$, a set \mathcal{F} of conditional constraints of possibly various types and two c -concepts ϕ, ψ of type i , if \mathcal{Q} is the set of all lex-minimal elements in the set of all $Q \subseteq P$, such that $T \cup Q \cup \mathcal{F} \cup \{(\phi|\top)[1, 1]\}$ is satisfiable, then

- if $\mathcal{Q} = \emptyset$, $\mathcal{F} \models_{\text{tight}}^{\text{lex}} (\psi|\phi)[1, 0]$ under PT ;
- if $\mathcal{Q} \neq \emptyset$, then $\mathcal{F} \models_{\text{tight}}^{\text{lex}} (\psi|\phi)[l, u]$ under PT , where $l = \min l'$ and $u = \max u'$, respectively, subject to $T \cup Q \cup \mathcal{F} \cup \{(\phi|\top)[1, 1]\} \models_{\text{tight}} (\psi|\top)[l', u']$ and $Q \in \mathcal{Q}$.

4.4.2 P- \mathcal{ALCI} Problems of Interest

Concerning the modified language P- \mathcal{ALCI} , whose syntax and semantics we presented in the previous subsection, we would like to revisit the following two computational problems, assuming that all reals in $[0, 1]$ considered are taken to be rational and denoting the set of rational numbers by \mathbb{Q} :

PTBOX CONSISTENCY (PTCON): Decide whether a given PTBox $PT = \langle T, P \rangle$ is consistent.

TIGHT LEXICOGRAPHIC ENTAILMENT (TLEXENT): Given a PTBox $PT = \langle T, P \rangle$, a finite set \mathcal{F} of conditional constraints (of possibly various types) and two c -concepts ϕ and ψ of type i , compute $l, u \in [0, 1] \cap \mathbb{Q}$, such that $\mathcal{F} \models_{\text{tight}}^{\text{lex}} (\psi|\phi)[l, u]$ under PT .

Lukasiewicz [64] shows that the problems PTCON and TLEXENT for his version of the language P- \mathcal{ALCI} (in fact for his languages P- $\mathcal{SHIF}(\mathbf{D})$ and P- $\mathcal{SHOIN}(\mathbf{D})$, based on the (non-probabilistic) description logics $\mathcal{SHIF}(\mathbf{D})$ and $\mathcal{SHOIN}(\mathbf{D})$, respectively, which are significantly more expressive than \mathcal{ALCI}) can be reduced to the following two problems:

SATISFIABILITY (SAT): Given an \mathcal{ALCI} knowledge base T and a finite set \mathcal{F} of conditional constraints, decide whether $T \cup \mathcal{F}$ is satisfiable.

TIGHT LOGICAL ENTAILMENT (TLOGENT): Given an \mathcal{ALCI} knowledge base T , a finite set \mathcal{F} of conditional constraints and a c -concept ψ , compute $l, u \in [0, 1] \cap \mathbb{Q}$, such that $T \cup \mathcal{F} \models_{\text{tight}} (\psi|\top)[l, u]$.

In fact, Lukasiewicz presents in Section 5.2 of [64] algorithms that reduce both PTCON and TLEXENT for his description logics P- $\mathcal{SHIF}(\mathbf{D})$ and P- $\mathcal{SHOIN}(\mathbf{D})$ to algorithms for SAT and TLOGENT for the languages P- $\mathcal{SHIF}(\mathbf{D})$ and P- $\mathcal{SHOIN}(\mathbf{D})$, respectively. Some of his ideas are borrowed from [63] and [43]. His algorithms apply also to the modified language P- \mathcal{ALCI} , that we presented in the previous subsection, that contains neither ABoxes nor PABoxes. We emphasize, however, that satisfiability for the modified P- \mathcal{ALCI} language refers to the existence of a model that satisfies the additional stipulation concerning satisfiability of all $\top_i, i \in V$, and the modified semantics. The following result allows us to estimate the

number of instances of SAT and TLOGENT that one has to solve in order to obtain solutions of given instances of PTCON and TLEXENT.

Theorem 51 (Theorem 5.4 (a) and (c) of [64]) *(a) An algorithm that solves PTCON uses $O(|P|^2)$ instances of SAT.*

(b) An algorithm that solves TLEXENT uses $O(2^{|P|})$ instances of SAT and TLOGENT.

SAT and TLOGENT, on the other hand, can be handled by reductions to deciding TBox satisfiability in \mathcal{ALCI} , deciding the solvability of systems of linear constraints and computing the optimal value of linear programs exactly as is the case for SAT and TLOGENT for P-SHIF(**D**) and P-SHOIN(**D**) in [64].

As far as SAT is concerned, it is reducible to deciding TBox satisfiability in \mathcal{ALCI} and whether a system of linear constraints is solvable. First, the set of possible worlds $R = \{I \in \mathcal{I}_C : I \models T\}$ is computed, using satisfiability $|\mathcal{I}_C|$ times to decide whether there exists a model \mathcal{I} of T , such that $\bigcap_{C \in I_i} C^{\mathcal{I}} \cap \bigcap_{C \in \mathcal{C}_i \setminus I_i} \Delta^{\mathcal{I}} \setminus C^{\mathcal{I}} \neq \emptyset$, for all $i \in V$. Then the following result, an analog of Theorem 5.5 of [64] for our modified language, applies:

Theorem 52 *Let T be an \mathcal{ALCI} TBox and \mathcal{F} a finite set of conditional constraints of possibly various types. Let $R = \{I \in \mathcal{I}_C : I \models T\}$. Then $T \cup \mathcal{F}$ is satisfiable iff the system of linear constraints*

$$\begin{cases} \sum_{r \in R, r \models \neg \psi \sqcap \phi} -ly_r + \sum_{r \in R, r \models \psi \sqcap \phi} (1-l)y_r \geq 0, & (\psi|\phi)[l, u] \in \mathcal{F}, l > 0 \\ \sum_{r \in R, r \models \neg \psi \sqcap \phi} uy_r + \sum_{r \in R, r \models \psi \sqcap \phi} (u-1)y_r \geq 0, & (\psi|\phi)[l, u] \in \mathcal{F}, u < 1 \\ \sum_{r \in R} y_r = 1, \quad y_r \geq 0, r \in R. \end{cases} \quad (4.3)$$

over the variables $y_r, r \in R$, is solvable.

As far as TLOGENT is concerned, it is reducible to deciding TBox satisfiability in \mathcal{ALCI} and computing the optimal values of two linear programs. The following result, an analog of Theorem 5.7 of [64], details the situation.

Theorem 53 *Let T be an \mathcal{ALCI} TBox, \mathcal{F} a finite set of conditional constraints of possibly various types and ψ a c -concept of type i . Assume that $T \cup \mathcal{F}$ is satisfiable and let $R = \{I \in \mathcal{I}_{\mathcal{C}} : I \models T\}$. Then l and u , such that $T \cup \mathcal{F} \models_{\text{tight}} (\psi|\top)[l, u]$ are given, respectively, by the optimal values of the following linear programs over the variables $y_r, r \in R$:*

$$\begin{aligned} & \text{minimize } \sum_{r \in R, r \models \psi} y_r \text{ subject to the linear constraints (4.3)} \\ & \text{maximize } \sum_{r \in R, r \models \psi} y_r \text{ subject to the linear constraints (4.3).} \end{aligned}$$

It should be fairly obvious from the work accomplished in [64] (see Theorems 6.3 and 6.4) that the following result applies concerning the complexity of the problems SAT,PTCON, on the one hand, and TLOGENT, TLEXENT, on the other:

Theorem 54 (a) SAT and PTCON are in EXP when $T \cup \mathcal{F}$ and PT, respectively, are defined in $P\text{-}\mathcal{ALCI}$;
 (b) TLOGENT and TLEXENT are in FEXP, when $T \cup \mathcal{F}$ and $\text{PT} \cup \mathcal{F}$, respectively, are defined in $P\text{-}\mathcal{ALCI}$.

4.5 Reduction from Federated to Unimodule Problems

We start this section by introducing the computational problems that are of interest when reasoning with a $\text{PF-}\mathcal{ALCI}$ knowledge base. We assume, once more that all reals in $[0, 1]$ considered are rational. Following [64], we would like to study the following decision and computation problems in the framework of $\text{PF-}\mathcal{ALCI}$ knowledge bases:

FEDERATED PTBOX CONSISTENCY (FPTCON): Decide whether a given PTBox $\text{PT} = \{\text{PT}_i\}_{i \in V}$, with $\text{PT}_i = \langle T_i, P_i \rangle$, is consistent.

FEDERATED TIGHT LEXICOGRAPHIC ENTAILMENT (FTLEXENT): Given a PTBox $\text{PT} = \{\text{PT}_i\}_{i \in V}$, with $\text{PT}_i = \langle T_i, P_i \rangle$, a finite collection $\mathcal{F} = \{\mathcal{F}_i\}_{i \in V}$ of sets of conditional constraints and two i -classification concepts ϕ and ψ , compute $l, u \in [0, 1] \cap \mathbb{Q}$, such that $\mathcal{F} \models_i^{\text{tlex}} (\psi|\phi)[l, u]$

under PT.

Our strategy for solving these two problems is to reduce them to the corresponding unimodule problems PTCON and TLEXENT for the language P- \mathcal{ALCI} , that were introduced in the previous section. Since algorithms for solving these problems can be easily extracted as modifications of the algorithms for the corresponding problems for the languages P- $\mathcal{SHIF}(\mathbf{D})$ and P- $\mathcal{SHOIN}(\mathbf{D})$, presented in [64], our reduction will provide a solution for the federated case. Moreover, based on the complexities of the unimodule problems, we can provide estimates for the corresponding complexities of the federated versions.

More precisely, we provide a reduction of a given instance of the federated PTBox consistency or the federated tight lexicographic entailment problem to an instance of the corresponding unimodule problem. Then, we apply the method of Lukasiewicz, using the algorithms involving SAT and TLOGENT, to obtain a solution for the original problem. Our task is to show how, given an instance α of either the FPTCON or the FTLEXENT problem, we can obtain an instance α^s of the corresponding problem for P- \mathcal{ALCI} , such that

- for FPTCON, α is consistent if and only if α^s is consistent and
- for FTLEXENT, $l, u \in [0, 1] \cap \mathbb{Q}$ are solutions of α if and only if l, u are solutions of α^s .

This will prove the decidability of FPTCON and FTLEXENT and will allow us to draw conclusions on their complexities based on the complexities of the corresponding problems for P- \mathcal{ALCI} .

Let $\text{PT} = \{\text{PT}_i\}_{i \in V}$ be a PF- \mathcal{ALCI} PTBox, with $\text{PT}_i = \langle T_i, P_i \rangle$. Taking after a similar construction, presented in [97], we construct a unimodule PTBox $\text{PT}^s = \langle T^s, P^s \rangle$ as follows:

The signature of PT^s is the union of the local signatures of the modules together with a global top \top , a global bottom \perp , local top concepts \top_i , for all $i \in V$, and, finally, a collection of new role names $\{R_{ij}\}_{(i,j) \in E}$, whose extensions in P- \mathcal{ALCI} interpretations will be used, roughly speaking, to simulate the image-domain relations of the federated interpretations. Formally,

$$\text{Sig}(\text{PT}^s) = \bigcup_i (\mathcal{C}_i \cup \mathcal{R}_i) \cup \{\top, \perp\} \cup \{\top_i : 1 \leq i \leq n\} \cup \{R_{ij} : (i, j) \in E\}.$$

To construct the unimodule TBox T^s and the unimodule PBox P^s , given the federated PTBox PT, we first introduce a mapping $\#_i$, which translates i -concept expressions C of the federated instance to concept expressions $\#_i(C)$ of the unimodule counterpart, and serves to maintain the compatibility of the concept domains. It is defined by induction on the structure of $C \in \widehat{\mathcal{C}}_i$ (for $i = j$, R_{ij} is assumed to be interpreted as the identity on Δ^i in any interpretation and, as a result, may be omitted from the following translation):

- $\#_i(C) = C$, if $C \in \mathcal{C}_i$;
- $\#_i(C) = \exists R_{ji}^-. \#_j(C)$, if $C \in \mathcal{C}_j \cap \widehat{\mathcal{C}}_i$;
- $\#_i(\neg_j D) = \exists R_{ji}^-. (\neg \#_j(D) \sqcap \top_j)$;
- $\#_i(D \boxplus_j E) = \exists R_{ji}^-. (\#_j(D) \boxplus \#_j(E))$, where $\boxplus = \sqcap$ or $\boxplus = \sqcup$;
- $\#_i(\exists_j R.D) = \exists R_{ji}^-. (\exists R_{kj}^-. (\exists R. (\exists R_{kj}. \#_j(D))))$, for $R \in \mathcal{R}_k \cup \mathcal{R}_k^-$;
- $\#_i(\forall_j R.D) = \exists R_{ji}^-. (\forall R_{kj}^-. (\forall R. (\forall R_{kj}. \#_j(D))))$, for $R \in \mathcal{R}_k \cup \mathcal{R}_k^-$.

Having defined $\#_i$, we show how various axioms derived from the structure of PT are added to T^s :

- For each $C \in \mathcal{C}_i$, $C \sqsubseteq \top_i$ is added to T^s .
- For each $R \in \mathcal{R}_i$, \top_i is stipulated to be the domain and range of R , i.e., $\top \sqsubseteq \forall R^-. \top_i$ and $\top \sqsubseteq \forall R. \top_i$ are added to T^s .
- For each new role name R_{ij} , \top_i is stipulated to be its domain and \top_j to be its range, i.e., $\top \sqsubseteq \forall R_{ij}^-. \top_i$ and $\top \sqsubseteq \forall R_{ij}. \top_j$ are added to T^s .
- For each $C \sqsubseteq D \in T_i$, $\#_i(C) \sqsubseteq \#_i(D)$ is added to T^s .

Finally, various axioms derived from the conditional constraints in PT, using the transformations $\#_i, i \in V$, are added to P^s :

- For every basic i -classification concept ϕ , the concept expression $\#_i(\phi)$ is added as a basic classification concept of type i of PT^s . Moreover, \top_i is declared to be a basic

classification concept of type i of PT^s . This defines the collection $\mathcal{B}^s = \{\mathcal{B}_i^s\}_{i \in V}$ of sets \mathcal{B}_i^s of basic classification concepts of type i of PT^s .

- For each i -conditional constraint $(\psi|\phi)[l, u]$ in P_i , $(\#_i(\psi)|\#_i(\phi))[l, u]$ is added to P^s .

Applying these definitions, we may obtain an instance of PTCON , given an instance of FPTCON . On the other hand, considering instances of the problem FTLEXENT , the collection $\mathcal{F} = \{\mathcal{F}_i\}_{i \in V}$ of sets of conditional constraints is translated to the collection \mathcal{F}^s by including, for every i -conditional constraint $(\psi|\phi)[l, u]$ in \mathcal{F}_i , the conditional constraint $(\#_i(\psi)|\#_i(\phi))[l, u]$ of type i in \mathcal{F}^s . We also translate the additional i -conditional constraint $(\psi|\phi)[l, u]$, that is given in the instance of the problem, to the conditional constraint $(\#_i(\psi)|\#_i(\phi))[l, u]$ of type i .

Example 7: In this example, we illustrate the reduction defined in this section by transforming the PTBox $\text{PT} = \langle \text{PT}_1, \text{PT}_2 \rangle$, defined in Examples 1 and 2 using the language $\text{PF-}\mathcal{ALCI}$, to the corresponding PTBox $\text{PT}^s = \langle T^s, P^s \rangle$ over the language $\text{P-}\mathcal{ALCI}$.

The general axioms included in T^s are:

$$\begin{array}{ll}
 \text{Insured} \sqsubseteq \top_1 & \text{Faculty} \sqsubseteq \top_2 \\
 \text{FullyInsured} \sqsubseteq \top_1 & \text{Lecturer} \sqsubseteq \top_2 \\
 \text{PartiallyInsured} \sqsubseteq \top_1 & \text{MaleLecturer} \sqsubseteq \top_2 \\
 & \text{FemaleLecturer} \sqsubseteq \top_2 \\
 \\
 \top \sqsubseteq \forall R_{12}^- . \top_1 & \top \sqsubseteq \forall R_{12} . \top_2
 \end{array}$$

The axioms in T^s that are induced by the axioms in T_1 and T_2 are, respectively,

$$\begin{array}{l}
 \text{FullyInsured} \sqsubseteq \text{Insured} \\
 \text{PartiallyInsured} \sqsubseteq \text{Insured} \\
 \text{PartiallyInsured} \sqsubseteq \neg \text{FullyInsured} \sqcap \top_1
 \end{array}$$

$$\text{MaleLecturer} \sqsubseteq \text{Lecturer}$$

$$\text{FemaleLecturer} = \text{Lecturer} \sqcap (\neg \text{MaleLecturer} \sqcap \top_2)$$

$$\text{Lecturer} \sqsubseteq \text{Faculty}$$

$$\text{Faculty} \sqsubseteq \exists R_{12}^-. \text{Insured}$$

Finally, the axioms in P^s that are induced by the axioms in P_1 and P_2 are, respectively,

$$(\text{FullyInsured} | \text{Insured})[0.8, 1]$$

$$(\text{HasDental} | \text{Insured})[1, 1]$$

$$(\neg \text{HasDental} \sqcap \top_1 | \text{PartiallyInsured})[1, 1]$$

$$(\text{DoesResearch} | \text{Faculty})[1, 1]$$

$$(\neg \text{DoesResearch} \sqcap \top_2 | \text{Lecturer})[1, 1]$$

$$(\exists R_{12}^-. \text{FullyInsured} | \text{Faculty})[1, 1]$$

$$(\exists R_{12}^-. \text{FullyInsured} | \text{Lecturer})[0.7, 1]$$

□

In the next section we show that the reduction \mathfrak{R} is sound and complete for both FPTCON and FTLEXENT. For FPTCON, this means that PT is consistent if and only if PT^s is consistent. On the other hand, for FTLEXENT, it means that $l, u \in [0, 1] \cap \mathbb{Q}$ are such that $\mathcal{F} \Vdash_i^{\text{tlex}} (\psi | \phi)[l, u]$ under PT if and only if they are such that $\mathcal{F}^s \Vdash_{\text{tight}}^{\text{lex}} (\#_i(\psi) | \#_i(\phi))[l, u]$ under PT^s .

4.6 Soundness and Completeness

4.6.1 Soundness and Completeness for FPTCon

In this section we present the soundness and completeness proofs of the translations from the federated problem FPTCON to the corresponding unimodule problem PTCON for P- \mathcal{ALCI} , the modified version of the problem studied for P- $\mathcal{SHOIN}(\mathbf{D})$ and P- $\mathcal{SHIF}(\mathbf{D})$ in [64]. More precisely, we aim to show that $\text{PT} = \{\text{PT}_i\}_{i \in V}$, with $\text{PT}_i = \langle T_i, P_i \rangle$, is consistent if and only if $\text{PT}^s = \langle T^s, P^s \rangle$ is consistent. To some degree, we rely on the proofs of soundness

and completeness of a reduction from F- \mathcal{ALCI} to \mathcal{ALCI} , that were presented in [97] (see also [98]).

Recall that a P- \mathcal{ALCI} knowledge base $\langle T, P \rangle$ is *consistent* if

- (i) T is satisfiable and
- (ii) there exists an ordered partition (P_0, \dots, P_k) of P , such that each P_i with $i \in \{0, 1, \dots, k\}$ is the set of all $F \in P \setminus (P_0 \cup \dots \cup P_{i-1})$ that are tolerated under T by $P \setminus (P_0 \cup \dots \cup P_{i-1})$.

On the other hand, as defined before, $PT = \{\langle T_i, P_i \rangle\}_{i \in V}$ is *consistent* if

- (i) $T = \{T_i\}_{i \in V}$ is consistent;
- (ii) There exist $k_i \geq 0, i \in V$, and ordered partitions $(P_0^i, \dots, P_{k_i}^i)$ of P_i , such that each P_j^i , $j = 0, \dots, k_i$, is the set of all $F \in P_i \setminus (P_0^i \cup \dots \cup P_{j-1}^i)$ that are tolerated under T by $P \setminus (P_0 \cup \dots \cup P_{j-1})$, where

$$P \setminus (P_0 \cup \dots \cup P_{j-1}) = \{P_i \setminus (P_0^i \cup \dots \cup P_{j-1}^i)\}_{i \in V}.$$

(In case, for some $i \in V$, $j > k_i$, we set $P_j^i = \emptyset$.)

Suppose that $T = \{T_i\}_{i \in V}$ is a F- \mathcal{ALCI} knowledge base, $\mathcal{B} = \{\mathcal{B}_i\}_{i \in V}$ a collection of sets of basic classification concepts, and $\mathcal{F} = \{\mathcal{F}_i\}_{i \in V}$ a collection of sets of conditional constraints. Consider the \mathcal{ALCI} knowledge base T^s , the collection $\mathcal{B}^s = \{\mathcal{B}_i^s\}_{i \in V}$ of sets of basic classification concepts and the set of conditional constraints P^s .

To each world $I = \{I_i\}_{i \in V}$ in $\mathcal{I}_{\mathcal{B}}$ there corresponds a world $I^s = \{I_i^s\}_{i \in V}$, with $I_i^s = \{\#_i(\phi) : \phi \in I_i\}$, in $\mathcal{I}_{\mathcal{B}^s}$. Conversely, to every world $I = \{I_i\}_{i \in V}$ in $\mathcal{I}_{\mathcal{B}^s}$, there corresponds a world $I^d = \{I_i^d\}_{i \in V}$, with $I_i^d = \{\phi \in \mathcal{B}_i : \#_i(\phi) \in I_i\}$, in $\mathcal{I}_{\mathcal{B}}$. Obviously, by definition, $I^{ds} = I$, for every $I \in \mathcal{I}_{\mathcal{B}^s}$, and $I^{sd} = I$, for every $I \in \mathcal{I}_{\mathcal{B}}$.

Let $\text{Pr} : \mathcal{I}_{\mathcal{B}} \rightarrow [0, 1]$ be a probabilistic interpretation on $\mathcal{I}_{\mathcal{B}}$. Define $\text{Pr}^s : \mathcal{I}_{\mathcal{B}^s} \rightarrow [0, 1]$ by setting $\text{Pr}^s(I^s) = \text{Pr}(I)$, for all $I \in \mathcal{I}_{\mathcal{B}}$. Clearly, Pr^s is a probabilistic interpretation on $\mathcal{I}_{\mathcal{B}^s}$.

Lemma 55 *If $\Pr : \mathcal{I}_{\mathcal{B}} \rightarrow [0, 1]$ is a probabilistic model of $T \cup \mathcal{F}$, then $\Pr^s : \mathcal{I}_{\mathcal{B}^s} \rightarrow [0, 1]$ is a probabilistic model of $T^s \cup \mathcal{F}^s$. Moreover, for every basic i -classification concept $\phi \in \mathcal{B}_i$, we have $\Pr_i(\phi) = \Pr^s(\#_i(\phi))$.*

Proof:

The fact that $I \models T$ iff $I^s \models T^s$ follows from the Soundness and Completeness Theorem of [98]. Furthermore, we have

$$\begin{aligned} \Pr_i(\phi) &= \sum \{\Pr(I) : I \models_i \phi\} \\ &= \sum \{\Pr^s(I^s) : I^s \models \#_i(\phi)\} \\ &= \Pr^s(\#_i(\phi)). \end{aligned}$$

This also shows that, if I is a model of \mathcal{F} , then I^s is a model of \mathcal{F}^s . ■

Let $\Pr : \mathcal{I}_{\mathcal{B}^s} \rightarrow [0, 1]$ be a probabilistic interpretation on $\mathcal{I}_{\mathcal{B}^s}$. Define $\Pr^d : \mathcal{I}_{\mathcal{B}} \rightarrow [0, 1]$ by setting $\Pr^d(I^d) = \Pr(I)$, for all $I \in \mathcal{I}_{\mathcal{B}^s}$. Clearly, \Pr^d is a probabilistic interpretation on $\mathcal{I}_{\mathcal{B}}$.

Lemma 56 *If $\Pr : \mathcal{I}_{\mathcal{B}^s} \rightarrow [0, 1]$ is a probabilistic model of $T^s \cup \mathcal{F}^s$, then $\Pr^d : \mathcal{I}_{\mathcal{B}} \rightarrow [0, 1]$ is a probabilistic model of $T \cup \mathcal{F}$. Moreover, for every basic i -classification concept $\phi \in \mathcal{B}_i$, we have $\Pr_i^d(\phi) = \Pr(\#_i(\phi))$.*

Proof:

Very similar to the proof of Lemma 55. ■

We continue the process of proving the soundness and completeness of the reduction $\text{PT} \mapsto \text{PT}^s$ for FPTCON by showing that the notion of tolerance in the federated case and that in the unimodule case are very tightly related. The following lemma expresses this connection precisely.

Lemma 57 *A collection $\mathcal{F} = \{\mathcal{F}_i\}_{i \in V}$ of sets of conditional constraints tolerates an i -conditional constraint $(\psi|\phi)[l, u]$ under an F -ALCI knowledge base $T = \{T_i\}_{i \in V}$ if and only if \mathcal{F}^s tolerates $(\#_i(\psi)|\#_i(\phi))[l, u]$ under T^s .*

Proof:

Suppose, first, that $\mathcal{F} = \{\mathcal{F}_i\}_{i \in V}$ tolerates $(\psi|\phi)[l, u]$ under $T = \{T_i\}_{i \in V}$. Then $T \cup \mathcal{F}$ has a model $\text{Pr} : \mathcal{I}_{\mathcal{B}} \rightarrow [0, 1]$, that verifies $(\psi|\phi)[l, u]$, i.e., such that $\text{Pr}_i(\phi) = 1$ and $\text{Pr}_i(\psi) \in [l, u]$. Let $\mathcal{B}^s = \{\mathcal{B}_i^s\}_{i \in V}$ be the collection of sets of basic classification concepts of PT^s , on which \mathcal{F}^s is based. Then, by Lemma 55, the function $\text{Pr}^s : \mathcal{I}_{\mathcal{B}^s} \rightarrow [0, 1]$, defined by $\text{Pr}^s(I^s) = \text{Pr}(I)$, is a probabilistic model of $T^s \cup \mathcal{F}^s$. Under this model we have $\text{Pr}^s(\#_i(\phi)) = \text{Pr}_i(\phi) = 1$ and $\text{Pr}^s(\#_i(\psi)) = \text{Pr}_i(\psi) \in [l, u]$. Thus, \mathcal{F}^s tolerates $(\#_i(\psi)|\#_i(\phi))[l, u]$ under T^s .

Suppose, conversely, that \mathcal{F}^s tolerates the constraint of type i $(\#_i(\psi)|\#_i(\phi))[l, u]$ under T^s . Then, there exists a probabilistic model $\text{Pr} : \mathcal{I}_{\mathcal{B}^s} \rightarrow [0, 1]$ of $T^s \cup \mathcal{F}^s$, that verifies $(\#_i(\psi)|\#_i(\phi))[l, u]$. Then, by Lemma 56, the function $\text{Pr}^d : \mathcal{I}_{\mathcal{B}} \rightarrow [0, 1]$ is a probabilistic model of $T \cup \mathcal{F}$, such that $\text{Pr}_i^d(\phi) = \text{Pr}(\#_i(\phi)) = 1$ and $\text{Pr}_i^d(\psi) = \text{Pr}(\#_i(\psi)) \in [l, u]$. Hence, \mathcal{F} tolerates $(\psi|\phi)[l, u]$ under T . \blacksquare

Taking into account the definitions concerning the relevant notions for $\text{P-}\mathcal{ALCI}$ and $\text{PF-}\mathcal{ALCI}$, to show that PT is consistent if and only if PT^s is consistent, it suffices to show the following:

Lemma 58 *There exist $k_i \geq 0, i \in V$, and ordered partitions $(P_0^i, \dots, P_{k_i}^i)$ of P_i , such that each $P_j^i, j = 0, \dots, k_i$, is the set of all $F \in P_i \setminus (P_0^i \cup \dots \cup P_{j-1}^i)$ that are tolerated under T by $P \setminus (P_0 \cup \dots \cup P_{j-1})$, where*

$$P \setminus (P_0 \cup \dots \cup P_{j-1}) = \{P_i \setminus (P_0^i \cup \dots \cup P_{j-1}^i)\}_{i \in V},$$

if and only if there exists an ordered partition (P_0^s, \dots, P_k^s) of P^s , such that each P_i^s with $i \in \{0, 1, \dots, k\}$ is the set of all $F \in P^s \setminus (P_0^s \cup \dots \cup P_{i-1}^s)$ that are tolerated under T^s by $P^s \setminus (P_0^s \cup \dots \cup P_{i-1}^s)$.

Proof:

Suppose, first, that there exist $k_i \geq 0, i \in V$, and ordered partitions $(P_0^i, \dots, P_{k_i}^i)$ of P_i , such that each $P_j^i, j = 0, \dots, k_i$, is the set of all $F \in P_i \setminus (P_0^i \cup \dots \cup P_{j-1}^i)$ that are tolerated under T by $P \setminus (P_0 \cup \dots \cup P_{j-1})$, where $P \setminus (P_0 \cup \dots \cup P_{j-1}) = \{P_i \setminus (P_0^i \cup \dots \cup P_{j-1}^i)\}_{i \in V}$. Let

$k := \max_{i \in V} k_i$ and set

$$P_j^s = \bigcup_{i \in V} \{(\#_i(\psi)|\#_i(\phi))[l, u] : (\psi|\phi)[l, u] \in P_j^s\},$$

for $j = 0, \dots, k$, where $P_j^i = \emptyset$, for all $j > k_i$. We must show that each P_i^s , $i \in \{0, 1, \dots, k\}$, is the set of all $F \in P^s \setminus (P_0^s \cup \dots \cup P_{i-1}^s)$ that are tolerated under T^s by $P^s \setminus (P_0^s \cup \dots \cup P_{i-1}^s)$. This follows from Lemma 57.

Conversely, assume there exists an ordered partition (P_0^s, \dots, P_k^s) of P^s , such that each P_i^s with $i \in \{0, 1, \dots, k\}$ is the set of all $F \in P^s \setminus (P_0^s \cup \dots \cup P_{i-1}^s)$ that are tolerated under T^s by $P^s \setminus (P_0^s \cup \dots \cup P_{i-1}^s)$. Then, set, for all $i \in V$ and all $j = 0, 1, \dots, k$,

$$P_j^i = \{(\psi|\phi)[l, u] \in P_i : (\#_i(\psi)|\#_i(\phi))[l, u] \in P_j^s\}.$$

Moreover, let $k_i = \max\{j : P_j^i \neq \emptyset\}$. Then, by Lemma 57, the ordered partitions $(P_0^i, \dots, P_{k_i}^i)$ of P_i are such that each P_j^i , $j = 0, \dots, k_i$, is the set of all $F \in P_i \setminus (P_0^i \cup \dots \cup P_{j-1}^i)$ that are tolerated under T by $P \setminus (P_0 \cup \dots \cup P_{j-1})$, where $P \setminus (P_0 \cup \dots \cup P_{j-1}) = \{P_i \setminus (P_0^i \cup \dots \cup P_{j-1}^i)\}_{i \in V}$. ■

Lemma 58 immediately yields

Theorem 59 (Soundness and Completeness of \mathfrak{R} for PTCon) *Let $\text{PT} = \{\text{PT}_i\}_{i \in V}$ be a PF- \mathcal{ALCI} knowledge base. PT is consistent if and only if PT^s is consistent.*

4.6.2 Soundness and Completeness for FTLexEnt

In this subsection we prove the soundness and completeness of the translation from the federated FTLEXENT problem to the corresponding problem for P- \mathcal{ALCI} . More precisely, we show that, given a PTBox $\text{PT} = \{\langle T_i, P_i \rangle\}_{i \in V}$, a finite collection $\mathcal{F} = \{\mathcal{F}_i\}_{i \in V}$ of sets of conditional constraints and two i -classification concepts ϕ and ψ , the set of rational numbers $l, u \in [0, 1] \cap \mathbb{Q}$, that satisfy $\mathcal{F} \Vdash_i^{\text{tlex}} (\psi|\phi)[l, u]$ under PT , is the same with that of the rational numbers $l, u \in [0, 1] \cap \mathbb{Q}$, satisfying $\mathcal{F}^s \Vdash_{\text{tight}}^{\text{lex}} (\#_i(\psi)|\#_i(\phi))[l, u]$ under PT^s . The fact that these

two subsets of \mathbb{Q} coincide shows that the algorithm developed in [64] for solving TLEXENT in the context of P-*SHIF*(**D**) and P-*SHOIN*(**D**), appropriately adjusted to solve TLEXENT for the modified language P-*ALCI*, that we presented in Section 4.4, may be used to also solve an instance of FTLEXENT. It also helps in providing a complexity estimate for the federated problem based on the corresponding complexity for the unimodule problem.

Given a finite collection $\mathcal{F} = \{\mathcal{F}_i\}_{i \in V}$ of sets of conditional constraints and an i -classification concept ϕ , let us define the collection $\mathcal{F}(\phi) = \{\mathcal{F}_j(\phi)\}_{j \in V}$, by

$$\mathcal{F}_j(\phi) = \begin{cases} \mathcal{F}_j, & \text{if } j \neq i \\ \mathcal{F}_i \cup \{(\phi|_{\top_i})[1, 1]\}, & \text{if } j = i \end{cases}$$

Moreover, in the unimodule setting, given a finite collection \mathcal{F} of conditional constraints, of possibly various types, and a classification concept ϕ of type i , let (see also [64])

$$\mathcal{F}(\phi) = \mathcal{F} \cup \{(\phi|_{\top})[1, 1]\}.$$

Note that $\mathcal{F}^s(\#_i(\phi)) = \mathcal{F}^s \cup \{(\#_i(\phi)|_{\top})[1, 1]\}$ whereas $\mathcal{F}(\phi)^s = \mathcal{F}^s \cup \{(\#_i(\phi)|_{\top_i})[1, 1]\}$. Because of the added stipulation adopted in our semantics that every interpretation of a federated ontology must have nonempty local domains and the fact that the extension of $\#_i(\phi)$ in any interpretation is a subset of the i -th domain, the interpretation of the two conditional constraints appearing in the singleton sets above coincide in any probabilistic model. More formally, in the following technical lemma it is shown that these two sets of conditional constraints have identical probabilistic models.

Lemma 60 *Let $\text{PT} = \{\text{PT}_i\}_{i \in V}$ be a PF-*ALCI* knowledge base, $\mathcal{F} = \{\mathcal{F}_i\}_{i \in V}$ a collection of sets of conditional constraints and ϕ an i -classification concept. A function $\text{Pr} : \mathcal{I}_{\mathcal{B}^s} \rightarrow [0, 1]$ is a probabilistic model of $\mathcal{F}^s(\#_i(\phi))$ iff it is a probabilistic model of $\mathcal{F}(\phi)^s$.*

Proof:

Taking into account the forms of the two sets of conditional constraints, the following string

of equalities proves the lemma:

$$\begin{aligned}
\Pr(\#_i(\phi)|\top) &= \frac{\Pr(\#_i(\phi)\cap\top)}{\Pr(\top)} \\
&= \Pr(\#_i(\phi)) \\
&= \frac{\Pr(\#_i(\phi)\cap\top_i)}{\Pr(\top_i)} \\
&= \Pr(\#_i(\phi)|\top_i).
\end{aligned}$$

The third equality follows from the fact that the interpretation of $\#_i(\phi)$ is stipulated to be a subset of the interpretation of \top_i in any model of the translation of PT and from the fact that our models are assumed to have nonempty local domains. ■

In Lemmas 61 and 62 it is shown that a probabilistic interpretation is a lex-minimal model of a federated knowledge base iff its unimodule counterpart is a lex-minimal model of the reduction of the federated knowledge base. This is the last auxiliary step on the road to proving Theorems 63 and 64, which establish the soundness and completeness of the given reduction from the problem FTLEXENT for a PF- \mathcal{ALCI} PTBox to the problem TLEXENT for a P- \mathcal{ALCI} PTBox.

Lemma 61 *Suppose that $T = \{T_i\}_{i \in V}$ is a F- \mathcal{ALCI} knowledge base and $\mathcal{F} = \{\mathcal{F}_i\}_{i \in V}$ a collection of finite sets of conditional constraints. If $\Pr : \mathcal{I}_{\mathcal{B}} \rightarrow [0, 1]$ is a lex-min model of $T \cup \mathcal{F}$, then $\Pr^s : \mathcal{I}_{\mathcal{B}^s} \rightarrow [0, 1]$ is a lex-min model of $T^s \cup \mathcal{F}^s$.*

Proof:

We prove the statement by contraposition. Suppose that $\Pr^s : \mathcal{I}_{\mathcal{B}^s} \rightarrow [0, 1]$ is not a lex-min model of $T^s \cup \mathcal{F}^s$. Thus, there exists a model $\Pr' : \mathcal{I}_{\mathcal{B}^s} \rightarrow [0, 1]$ of $T^s \cup \mathcal{F}^s$ that is lex-preferable to \Pr^s . But then, by Lemma 55 and the definitions of lex-preference, $\Pr'^d : \mathcal{I}_{\mathcal{B}} \rightarrow [0, 1]$ is a model of $T \cup \mathcal{F}$, which is lex-preferable to $\Pr^{sd} = \Pr$. This shows that $\Pr : \mathcal{I}_{\mathcal{B}} \rightarrow [0, 1]$ is not a lex-min model of $T \cup \mathcal{F}$. ■

Lemma 62 *Suppose that $T = \{T_i\}_{i \in V}$ is a F- \mathcal{ALCI} knowledge base and $\mathcal{F} = \{\mathcal{F}_i\}_{i \in V}$ a collection of finite sets of conditional constraints. If $\Pr : \mathcal{I}_{\mathcal{B}^s} \rightarrow [0, 1]$ is a lex-min model of $T^s \cup \mathcal{F}^s$, then $\Pr^d : \mathcal{I}_{\mathcal{B}} \rightarrow [0, 1]$ is a lex-min model of $T \cup \mathcal{F}$.*

Proof:

Very similar to the proof of Lemma 61. ■

Theorem 63 *Suppose that $\text{PT} = \{\langle T_i, P_i \rangle\}_{i \in V}$ is a PF- \mathcal{ALCI} knowledge base, $\mathcal{F} = \{\mathcal{F}_i\}_{i \in V}$ a collection of finite sets of conditional constraints and ϕ, ψ two i -classification concepts. Then, $\mathcal{F} \models_i^{\text{lex}} (\psi|\phi)[l, u]$ under PT if and only if $\mathcal{F}^s \models^{\text{lex}} (\#_i(\psi)|\#_i(\phi))[l, u]$ under PT^s .*

Proof:

Suppose, first, that $\mathcal{F} \models_i^{\text{lex}} (\psi|\phi)[l, u]$ under PT . Then $\text{Pr}_i(\psi) \in [l, u]$, for every lex-min model Pr of $T \cup \mathcal{F}(\phi)$. Assume, for the sake of obtaining a contradiction, that $\mathcal{F}^s \not\models^{\text{lex}} (\#_i(\psi)|\#_i(\phi))[l, u]$ under PT^s . Thus, there exists a lex-min model Pr' of $T^s \cup \mathcal{F}(\phi)^s$, such that $\text{Pr}'(\#_i(\psi)) \notin [l, u]$. By Lemmas 56, 60 and 62, Pr'^d is a lex-min model of $T \cup \mathcal{F}(\phi)$, such that $\text{Pr}'^d(\psi) = \text{Pr}'^{ds}(\#_i(\psi)) = \text{Pr}'(\#_i(\psi)) \notin [l, u]$. But this is a contradiction.

The proof of the converse statement is very similar, but uses Lemmas 55 and 61 in place of Lemmas 56 and 62, respectively. ■

Theorem 64 *Let $\text{PT} = \{\langle T_i, P_i \rangle\}_{i \in V}$ be a PF- \mathcal{ALCI} PTBox, $\mathcal{F} = \{\mathcal{F}_i\}_{i \in V}$ a collection of finite sets of conditional constraints and ϕ, ψ two i -classification concepts. Then, it is the case that $\mathcal{F} \models_i^{\text{tlex}} (\psi|\phi)[l, u]$ under PT if and only if $\mathcal{F}^s \models_{\text{tight}}^{\text{lex}} (\#_i(\psi)|\#_i(\phi))[l, u]$ under PT^s .*

Proof:

Obvious from Theorem 63. ■

4.6.3 Algorithmic Significance and Complexities

In this subsection, we examine the significance of the reductions from the federated problems FPTCON and FTLEXENT to the corresponding unimodule problems, that were presented in the previous subsections. In Section 4.4, it was shown, based on the algorithms provided by Lukasiewicz [64] for the expressive probabilistic description logics $\text{P-SHIF}(\mathbf{D})$ and $\text{P-SHOLN}(\mathbf{D})$, that both PTCON and TLEXENT are decidable for our version of the language

P- \mathcal{ALCI} . Thus, our reductions show that both problems FPTCON and FTLEXENT for PF- \mathcal{ALCI} are also decidable. In fact, it is not much more difficult, based on Theorems 59 and 64, to show, using Theorems 6.3 and 6.4 of [64], that FPTCON is complete for exponential time and that FTLEXENT is complete for FEXP, the class corresponding to exponential time for problems that output a value. We formally state these results in the next theorem:

Theorem 65 *FPTCON is complete for EXP. FTLEXENT is complete for FEXP.*

Proof:

Since FPTCON and FTLEXENT were reduced in Theorems 59 and 64, respectively, to the corresponding unimodule problems in polynomial time, and, by Theorem 54 (an adaptation of Theorems 6.3 and 6.4 of [64]), these problems are in EXP and FEXP, respectively, we know that FPTCON and FTLEXENT are in EXP and FEXP, respectively. Hardness for both problems is inherited from the fact that deciding the satisfiability of a knowledge base with arbitrary TBoxes in \mathcal{ALCI} is complete for EXP (see [84, 95]). ■

4.7 Summary

In this paper, inspired by the work of Lukasiewicz on expressive probabilistic description logics [64], we have introduced the federated probabilistic description language PF- \mathcal{ALCI} . This language is, to the best of our knowledge, the first language presented in the literature that combines three desirable features:

- modularity, so as to support autonomous but interrelated ontology development in the semantic web;
- contextualization of all logical connectives so that meaning depends on the module where a definition is provided;
- probabilistic features that support probabilistic terminological and default terminological reasoning.

Probabilistic treatment of both terminological and assertional knowledge was presented for the description logics $P\text{-}\mathcal{SHIF}(\mathbf{D})$ and $P\text{-}\mathcal{SHOIN}(\mathbf{D})$ in [64]. On the other hand, a federated description logic $F\text{-}\mathcal{ALCI}$, based on \mathcal{ALCI} , supporting contextual connectives, was introduced in [97]. Since in the present work, we make a first attempt at integrating all these features into a single language, we opted to keep the language rather simple. Instead of a more expressive description logic, we based our language on \mathcal{ALCI} and chose to deal only with TBoxes and probabilistic terminological and terminological default knowledge rather than incorporating also ABoxes and assertional probabilistic statements. These extensions, which are very desirable for obvious reasons, are left as goals for future work.

Chapter 5. F- \mathcal{ALCIK} : FULLY CONTEXTUALIZED FEDERATED \mathcal{ALCI} WITH EPISTEMIC OPERATORS

Abstract

Many semantic web applications require support for knowledge representation and inference over a federation of multiple autonomously developed ontology modules allowing selective partial reuse of knowledge, based on a well-coordinated context-sensitive semantics. Federated \mathcal{ALCIK} or F- \mathcal{ALCIK} is a modular description logic, each of whose modules is roughly an \mathcal{ALCIK} ontology (\mathcal{ALC} with inverse roles and with an epistemic operator). F- \mathcal{ALCIK} supports importing of both concepts and roles across modules and contextualized interpretation of all logical connectives. Moreover, it allows non-monotonic reasoning, formalization of procedural rules as well as various forms of sophisticated query formulation, as does its single-module counterpart \mathcal{ALCIK} . To show that this contextualized epistemic federated description logic is decidable and to evaluate the complexity of the corresponding satisfiability problem, a sound and complete reduction to the description logic \mathcal{ALCIK} , modeled after a corresponding reduction from F- \mathcal{ALCI} to \mathcal{ALCI} , is provided.

5.1 Introduction

This paper belongs to the tradition of exploring various description logics as the foundation for implementing ontology languages to represent ontologies in the semantic web.

Whereas the actual world-wide web deals mostly with information that is geared towards human understanding and consumption, the vast amounts of information that have recently been accumulating in the web require that they be amenable to processing by machines. The

effort to usher this new era of machine interpretable and machine interoperable resources and services in the web has led to the paradigm of the *semantic web* [11]. In the semantic web, information is organized in the form of ontologies and, most often, ontologies are constructed using ontology languages, such as OIL and OWL [50]. Almost all ontology languages have as their theoretical foundation Description Logics (DLs) [17, 18], a family of logic-based knowledge representation languages, that are decidable fragments of first-order logic or decidable extensions of those fragments with various added constructs [2].

Non-monotonic reasoning and procedural rules, which were among many of the desired features of frame-based-systems, cannot be directly expressed in fragments of first-order logic. To accommodate these features in DLs, Donini et al. introduced in [35] an *epistemic operator* on top of the DL \mathcal{ALC} (see [85]). This operator effectively extends \mathcal{ALC} to a fragment of first-order non-monotonic modal logic. It achieves the goal of formalizing procedural rules and, in addition, enables more sophisticated forms of query formulation. Among these are various forms of closed-world reasoning. The knowledge operator in [35] is interpreted as in [79, 59]. This enables the queries to refer both to aspects of the external world, as represented by the knowledge base, and to aspects of what the knowledge base knows about the external world.

Ontologies that refer to one big domain of discourse are ordinarily developed by autonomous independent groups with specialized knowledge of several different, but partially overlapping, subdomains of the entire domain. In each of these independent and autonomously developed modules, the terminology and information is context specific and context sensitive. Moreover, coordination is needed so that concepts and terms may be partially reused by various developers working in interrelated contexts, without having to be explicitly redefined or rewritten. These requirements point to the necessity of introducing into terminological languages two additional features that are not present in basic DLs. One is *modularity*, i.e., the ability to develop independently autonomous but interrelated modules, whose concepts can be easily reused, and the other is *contextualization*, i.e., the property that a statement or a query takes a special meaning depending on the context in which it appears or is posed, respectively. Various modular description logics have appeared in the literature to deal with the first requirement.

Among the best known are Distributed Description Logics (DDL) [14], \mathcal{E} -Connections [26, 56] and Package-Based Description Logics (P-DLs) [8]. As far as context-sensitivity is concerned, it was recognized early in artificial intelligence by McCarthy and his collaborators (see, e.g., [69, 20]). Significant effort has been subsequently invested in contextualizing ontologies (in the non modular setting) [15, 45]. The recently introduced federated description logic $F\text{-}\mathcal{ALCI}$ [97, 98] addresses both the need for modular development of ontologies and the need for contextualization. In fact, it has been the first attempt to contextualize all logical connectives and not just logical negation, as had been the case in previously introduced modular DLs.

In the present work, an attempt is made to combine all three desirable features that were mentioned above, namely,

- expressibility of non-monotonic aspects of reasoning and of procedural rules by incorporating epistemic knowledge operators into the language,
- modularity to allow independent autonomous development of coordinated ontology modules and reuse of knowledge across modules and
- contextualization of both the logical connectives and the epistemic concepts and roles,

in a single modular ontology language. This effort builds on previous work on the fully contextualized federated language $F\text{-}\mathcal{ALCI}$ [97], which does not contain epistemic operators. The resulting language $F\text{-}\mathcal{ALCIK}$ is, to the best of our knowledge, the first language that combines modularity, contextualization and epistemic operators. When only one module is present, $F\text{-}\mathcal{ALCIK}$ reduces to the ordinary DL \mathcal{ALCIK} , which is an extension with inverse roles of the language \mathcal{ALCK} , that was studied in detail in [35]. We introduce a reduction of a given $F\text{-}\mathcal{ALCIK}$ knowledge base to an ordinary \mathcal{ALCIK} ontology, which can be performed in polynomial time in the size of the ontology. This reduction is shown to be sound and complete, in the sense that the federated ontology is consistent from a specific module's point of view if the corresponding top concept is satisfiable in the reduced uni-module ontology and vice-versa. Soundness and completeness, together with known decidability and complexity results for \mathcal{ALCIK} [35] allow us to obtain corresponding results for $F\text{-}\mathcal{ALCI}$. More precisely, it is

shown that the consistency problem for a $F\text{-}\mathcal{ALCIK}$ knowledge base is decidable and that it is EXPTIME-complete.

The paper is organized as follows: In Section 5.2 we present the syntax and the semantics of the fully contextualized, federated epistemic language $F\text{-}\mathcal{ALCIK}$. The main features are illustrated using some examples. In Section 5.3, we present a reduction from the federated language $F\text{-}\mathcal{ALCIK}$ to its single module counterpart \mathcal{ALCIK} . It serves to demonstrate the decidability of the consistency problem for $F\text{-}\mathcal{ALCIK}$, based on the decidability of the corresponding problem in \mathcal{ALCIK} . An example serves to illustrate the reduction. Section 5.4 contains the main results concerning the soundness and completeness of the reduction. At the end of the section, some consequences of these results are also explored. Finally, in the Summary section, we review the main contributions of the paper.

5.2 The Federated Languages $F\text{-}\mathcal{ALCI}$ and $F\text{-}\mathcal{ALCIK}$

In this work, we use the prefix “F-”, standing for **F**ederated, to denote a contextualized federated language, limited to acyclic importing of concepts, roles and contextualized logical connectives (see [97]). Moreover, following [35], we append at the end the letter \mathcal{K} to denote inclusion of an epistemic operator (or many epistemic operators) in the language.

We present the syntax and the semantics of the language $F\text{-}\mathcal{ALCIK}$, which extends the language $F\text{-}\mathcal{ALCI}$, that was introduced in [97] (see also [98]) and for which a tableau-based algorithm was presented in [99], with the addition of epistemic operators. One operator \mathbf{K}_i is added for each module i and the intended extension of an epistemic concept $\mathbf{K}_i C$ in module i is supposed to be the set of all individuals in the i -th domain that are in the extension of C in all models of the distributed knowledge base under consideration. The framework assumes a fixed domain semantics, i.e., that the domains of all modules are fixed in all local interpretations, and a fixed image domain relation semantics, i.e., that the image domain relations across the various modules are also fixed in all federated interpretations under consideration.

5.2.1 The Syntax

Suppose a directed acyclic graph $G = \langle V, E \rangle$, with $V = \{1, 2, \dots, n\}$, is given. The intuition is that its n nodes correspond to local modules of a modular ontology and its edges correspond to the importing relations between these modules. For technical reasons, we add a loop on each vertex of G .

For every node $i \in V$, the signature of the i -language always includes a set \mathcal{C}_i of **i -concept names** and a set \mathcal{R}_i of **i -role names**. We assume that all sets of names are pairwise disjoint. Out of these, a set of *i -concept expressions* $\widehat{\mathcal{C}}_i$ of F- \mathcal{ALCI} and $\widetilde{\mathcal{C}}_i$ of F- \mathcal{ALCIK} and a set of *i -role expressions* $\widehat{\mathcal{R}}_i$ of F- \mathcal{ALCI} and $\widetilde{\mathcal{R}}_i$ of F- \mathcal{ALCIK} are built.

Recall that the description logic \mathcal{ALCI} allows concept expressions that are constructed recursively from its signature symbols, i.e., its role and concept names, using negation, conjunction, disjunction, value and existential restriction and inverses of role names. Its formulas are subsumptions between concept expressions.

The syntax of the description logic F- \mathcal{ALCI} and that of F- \mathcal{ALCIK} are defined as follows:

Definition 66 (Roles and Concepts) *The set of i -roles or i -role expressions $\widetilde{\mathcal{R}}_i$ of F- \mathcal{ALCIK} consists of expressions of the form*

$$R, \mathbf{K}_j R, \text{ where } R \in \mathcal{R}_j \cup \mathcal{R}_j^-, (j, i) \in E,$$

where $\mathcal{R}_j^- = \{R^- : R \in \mathcal{R}_j\}$. *The set of i -roles or i -role expressions $\widehat{\mathcal{R}}_i$ of F- \mathcal{ALCI} consists of those i -role expressions in $\widetilde{\mathcal{R}}_i$ that do not use the epistemic operators \mathbf{K}_j , $(j, i) \in E$.*

The set of i -concepts or i -concept expressions $\widetilde{\mathcal{C}}_i$ of F- \mathcal{ALCIK} is defined recursively as follows:

$$A \in \mathcal{C}_j, \top_j, \perp_j, \neg_j C, C \sqcap_j D, C \sqcup_j D, \exists_j R.C, \forall_j R.C, \mathbf{K}_j C, \quad (5.1)$$

where $(j, i) \in E$, $C, D \in \widetilde{\mathcal{C}}_i \cap \widetilde{\mathcal{C}}_j$ and $R \in \widetilde{\mathcal{R}}_i \cap \widetilde{\mathcal{R}}_j$.

The set of i -concepts or i -concept expressions $\widehat{\mathcal{C}}_i$ of F- \mathcal{ALCI} consists of those i -concept expressions in $\widetilde{\mathcal{C}}_i$ that do not use the epistemic operators \mathbf{K}_j , $(j, i) \in E$ (either in front of role expressions or in front of concept expressions).

Using the concepts and roles of F- \mathcal{ALCI} or F- \mathcal{ALCIK} , we define their formulas, as follows:

Definition 67 (Formulas) *The i -formulas of F- \mathcal{ALCI} are expressions of the form $C \sqsubseteq D$, with $C, D \in \widehat{\mathcal{C}}_i$, for all $i \in V$. Similarly, the i -formulas of F- \mathcal{ALCIK} are expressions of the form $C \sqsubseteq D$, with $C, D \in \widetilde{\mathcal{C}}_i$, for all $i \in V$.*

An F- \mathcal{ALCI} -**TBox** (F- \mathcal{ALCIK} -**TBox**) or, simply, **TBox**, when the relevant language is clear from context, is a collection $T = \{T_i\}_{i \in V}$, where T_i is a finite set of i -formulas of F- \mathcal{ALCI} (F- \mathcal{ALCIK} , respectively) for all $i \in V$, called the i -**TBox**. Since, in this paper, we do not consider RBoxes or ABoxes, the terms *TBox*, *ontology* and *knowledge base* (KB) will be used interchangeably.

We use, for every $i \in V$, the notation $\overline{\mathcal{R}}_i$ and $\overline{\mathcal{C}}_i$ to denote the set of i -roles and of i -concepts, respectively, that occur in T_i . These could be in $\widehat{\mathcal{R}}_i$ and $\widehat{\mathcal{C}}_i$, respectively, or $\widetilde{\mathcal{R}}_i$ and $\widetilde{\mathcal{C}}_i$, respectively, depending on the language used in T . $\overline{\mathcal{C}}_i$ is a finite subset of $\widehat{\mathcal{C}}_i$ or $\widetilde{\mathcal{C}}_i$, for every $i \in V$, and, similarly for roles. A role name in $\mathcal{R}_j \cap \overline{\mathcal{R}}_i$ or a concept name in $\mathcal{C}_j \cap \overline{\mathcal{C}}_i$ is said to be **imported from** module j **to** module i .

Example 1 (Syntax): Consider a federated knowledge base with two modules T_1 and T_2 over the directed graph consisting of two nodes 1 and 2 and a directed edge from 1 to 2. Let **Faculty**, **Course** be 1-concept names and **Student** a 2-concept name and, also, **teaches** be a 1-role name and **enrolled** a 2-role name. Assume that module T_1 consists of the formulas

$$\begin{aligned} \top_1 &\sqsubseteq \text{Faculty} \sqcup_1 \text{Course} \\ \text{Faculty} \sqcap_1 \text{Course} &\sqsubseteq \perp \\ \exists_1 \text{teaches.Course} &\sqsubseteq \text{Faculty} \end{aligned}$$

The first two formulas express the fact that the extensions of the 1-concept names are disjoint and cover the domain of discourse of the first module. The last formula expresses the fact that every individual of the local domain that teaches a course must be a faculty. Moreover, let T_2

consist of the formulas

$$\begin{aligned}
\top_2 &\sqsubseteq \text{Faculty} \sqcup_2 \text{Course} \sqcup_2 \text{Student} \\
\text{Faculty} \sqcap_2 \text{Course} &\sqsubseteq \perp \\
\text{Faculty} \sqcap_2 \text{Student} &\sqsubseteq \perp \\
\text{Course} \sqcap_2 \text{Student} &\sqsubseteq \perp \\
\exists_2 \text{enrolled.Student} &\sqsubseteq \text{Course}
\end{aligned}$$

Again, the first four formulas assert that the extensions of all concept names, as witnessed by the second module, are pairwise disjoint and cover the domain of discourse of the second module. The last formula, on the other hand, expresses the fact that every individual in the domain that has enrolled a student must be a course. Among the several questions that can be posed against this knowledge base, are

- (i) the one concerning the satisfiability of the concept expression

$$\exists_1 \text{teaches.Course} \sqcap_1 \neg_1 \mathbf{K}_1 \text{Faculty}$$

as witnessed by module T_1 and

- (ii) the one concerning the satisfiability of

$$\exists_2 \text{teaches}.\exists_2 \text{enrolled.Student} \sqcap_2 \neg_2 \text{Faculty}$$

as witnessed by T_2 .

5.2.2 The Semantics

In this subsection, we present the semantics for the language F- \mathcal{ALCIK} . To do this, we need to revisit the semantics of F- \mathcal{ALCI} .

Definition 68 (F- \mathcal{ALCI} Interpretation) An F- \mathcal{ALCI} interpretation

$$\mathcal{I} = \langle \{\mathcal{I}_i\}_{i \in V}, \{r_{ij}\}_{(i,j) \in E} \rangle$$

consists of a family $\mathcal{I}_i = \langle \Delta^i, \cdot^i \rangle, i \in V$, of **local interpretations**, together with a family of **image domain relations** $r_{ij} \subseteq \Delta^i \times \Delta^j, (i,j) \in E$, such that $r_{ii} = \text{id}_{\Delta^i}$, for all $i \in V$.

Notation: For a binary relation $r \subseteq \Delta^i \times \Delta^j$, $X \subseteq \Delta^i$ and $S \subseteq \Delta^i \times \Delta^i$, we set

$$r(X) := \{y \in \Delta^j : (\exists x \in X)((x, y) \in r)\},$$

$$r(S) := \{(z, w) \in \Delta^j \times \Delta^j : (\exists (x, y) \in S)((x, z), (y, w) \in r)\}.$$

A local interpretation function \cdot^i interprets i -role names and i -concept names, as well as \perp_i and \top_i , as follows:

- $C^i \subseteq \Delta^i$, for all $C \in \mathcal{C}_i$,
- $R^i \subseteq \Delta^i \times \Delta^i$, for all $R \in \mathcal{R}_i$,
- $\top_i^i = \Delta^i$, $\perp_i^i = \emptyset$.

The interpretations of imported role names and imported concept names are computed by the following rules:

- $C^i = r_{ji}(C^j)$, for all $C \in \mathcal{C}_j \cap \widehat{\mathcal{C}}_i$,
- $R^i = r_{ji}(R^j)$, for all $R \in \mathcal{R}_j \cap \widehat{\mathcal{R}}_i$,
- $\top_j^i = r_{ji}(\Delta^j)$, $\perp_j^i = \emptyset$.

The recursive features of the local interpretation function \cdot^i are as follows:

- $R^{-i} = R^{i-}$, for all $R \in \mathcal{R}_i$, where $R^{i-} = \{(y, x) : (x, y) \in R^i\}$
- $(\neg_j C)^i = r_{ji}(\Delta^j \setminus C^j)$

- $(C \sqcap_j D)^i = r_{ji}(C^j \cap D^j)$
- $(C \sqcup_j D)^i = r_{ji}(C^j \cup D^j)$
- $(\exists_j R.C)^i = r_{ji}(\{x \in \Delta^j : (\exists y)((x, y) \in R^j \text{ and } y \in C^j)\})$
- $(\forall_j R.C)^i = r_{ji}(\{x \in \Delta^j : (\forall y)((x, y) \in R^j \text{ implies } y \in C^j)\})$

In case more than one interpretations are under consideration, the superscript i will be augmented to \mathcal{I}_i to make explicit the interpretation that is referred to in a specific context.

Definition 69 (F-ALCIK Interpretation) *An F-ALCIK interpretation is a pair $(\mathcal{I}, \mathcal{W})$, where \mathcal{I} is a F-ALCI interpretation and \mathcal{W} is a set of F-ALCI interpretations, such that, for every $\mathcal{J} \in \mathcal{W}$ and every $i, j \in V$,*

- $\Delta^{\mathcal{J}_i} = \Delta^{\mathcal{I}_i}$, and
- $r_{ij}^{\mathcal{J}} = r_{ij}^{\mathcal{I}}$, if $(i, j) \in E$.

i.e., local domains corresponding to the same module across all interpretations in \mathcal{W} are assumed to be fixed and the same holds for the image domain relations. We define $\Delta^i := \Delta^{\mathcal{J}_i}$, $i \in V$, and $r_{ij} := r_{ij}^{\mathcal{I}}$, $(i, j) \in E$. Because of the two assumptions displayed above, these definitions are not ambiguous.

The pair $(\mathcal{I}, \mathcal{W})$ gives rise to a local interpretation function $\cdot^{\mathcal{I}_i, \mathcal{W}}$, that interprets i -role names and i -concept names, as well as \perp_i and \top_i , as follows:

- $C^{\mathcal{I}_i, \mathcal{W}} = C^{\mathcal{I}_i} \subseteq \Delta^i$, for all $C \in \mathcal{C}_i$,
- $R^{\mathcal{I}_i, \mathcal{W}} = R^{\mathcal{I}_i} \subseteq \Delta^i \times \Delta^i$, for all $R \in \mathcal{R}_i$,
- $\top_i^{\mathcal{I}_i, \mathcal{W}} = \Delta^i$, $\perp_i^{\mathcal{I}_i, \mathcal{W}} = \emptyset$.

The interpretations of imported role names and imported concept names are computed by the following rules:

- $C^{\mathcal{I}_i, \mathcal{W}} = r_{ji}(C^{\mathcal{I}_j, \mathcal{W}})$, for all $C \in \mathcal{C}_j \cap \tilde{\mathcal{C}}_i$,

- $R^{\mathcal{I}_i, \mathcal{W}} = r_{ji}(R^{\mathcal{I}_j, \mathcal{W}})$, for all $R \in \mathcal{R}_j \cap \tilde{\mathcal{R}}_i$,
- $\top_j^{\mathcal{I}_i, \mathcal{W}} = r_{ji}(\Delta^j)$, $\perp_j^{\mathcal{I}_i, \mathcal{W}} = \emptyset$.

The recursive features of the local interpretation function $\cdot^{\mathcal{I}_i, \mathcal{W}}$ are as follows:

- $R^{-\mathcal{I}_i, \mathcal{W}} = (R^{\mathcal{I}_i, \mathcal{W}})^-$, for all $R \in \tilde{\mathcal{R}}_i$
- $(\mathbf{K}_j R)^{\mathcal{I}_i, \mathcal{W}} = r_{ji}(\bigcap_{\mathcal{J} \in \mathcal{W}} R^{\mathcal{J}_j, \mathcal{W}})$, for all $R \in \tilde{\mathcal{R}}_i$
- $(\neg_j C)^{\mathcal{I}_i, \mathcal{W}} = r_{ji}(\Delta^j \setminus C^{\mathcal{I}_j, \mathcal{W}})$
- $(C \sqcap_j D)^{\mathcal{I}_i, \mathcal{W}} = r_{ji}(C^{\mathcal{I}_j, \mathcal{W}} \cap D^{\mathcal{I}_j, \mathcal{W}})$
- $(C \sqcup_j D)^{\mathcal{I}_i, \mathcal{W}} = r_{ji}(C^{\mathcal{I}_j, \mathcal{W}} \cup D^{\mathcal{I}_j, \mathcal{W}})$
- $(\exists_j R.C)^{\mathcal{I}_i, \mathcal{W}} = r_{ji}(\{x \in \Delta^j : (\exists y)((x, y) \in R^{\mathcal{I}_j, \mathcal{W}} \text{ and } y \in C^{\mathcal{I}_j, \mathcal{W}})\})$
- $(\forall_j R.C)^{\mathcal{I}_i, \mathcal{W}} = r_{ji}(\{x \in \Delta^j : (\forall y)((x, y) \in R^{\mathcal{I}_j, \mathcal{W}} \text{ implies } y \in C^{\mathcal{I}_j, \mathcal{W}})\})$
- $(\mathbf{K}_j C)^{\mathcal{I}_i, \mathcal{W}} = r_{ji}(\bigcap_{\mathcal{J} \in \mathcal{W}} C^{\mathcal{J}_j, \mathcal{W}})$

Note that, for every $R \in \hat{\mathcal{R}}_i$ and every $C \in \hat{\mathcal{C}}_i$, we have that $R^{\mathcal{I}_i, \mathcal{W}} = R^{\mathcal{I}_i}$ and $C^{\mathcal{I}_i, \mathcal{W}} = C^{\mathcal{I}_i}$, i.e., this definition restricted to the F- \mathcal{ALCI} roles and concepts coincides with the one given previously for F- \mathcal{ALCI} interpretations, if we disregard the set \mathcal{W} .

For all $i \in V$, i -**satisfiability**, denoted by \models_i , is defined by $(\mathcal{I}, \mathcal{W}) \models_i C \sqsubseteq D$ iff $C^{\mathcal{I}_i, \mathcal{W}} \subseteq D^{\mathcal{I}_i, \mathcal{W}}$. Given a TBox $T = \{T_i\}_{i \in V}$, the interpretation $(\mathcal{I}, \mathcal{W})$ is a **model** of T_i , written $(\mathcal{I}, \mathcal{W}) \models_i T_i$, iff $(\mathcal{I}, \mathcal{W}) \models_i \tau$, for every $\tau \in T_i$. Moreover, $(\mathcal{I}, \mathcal{W})$ is a **model** of T , written $(\mathcal{I}, \mathcal{W}) \models T$, iff $(\mathcal{I}, \mathcal{W}) \models_i T_i$, for every $i \in V$. An **epistemic model** of T is a maximal nonempty set \mathcal{W} of interpretations, such that, for every $\mathcal{I} \in \mathcal{W}$, $(\mathcal{I}, \mathcal{W}) \models T$.

Let $w \in V$. Define $G_w = \langle V_w, E_w \rangle$ to be the subgraph of G induced by those vertices in G from which w is reachable and $T_w^* := \{T_i\}_{i \in V_w}$. We say that an F- \mathcal{ALCIK} -ontology $T = \{T_i\}_{i \in V}$ is **consistent as witnessed by a module** T_w if T_w^* has an epistemic model $\mathcal{W} = \{\mathcal{I}\}$, with $\mathcal{I} = \langle \{\mathcal{I}_i\}_{i \in V_w}, \{r_{ij}\}_{(i,j) \in E_w} \rangle$, such that $\Delta^w \neq \emptyset$. A concept C is **satisfiable as witnessed by** T_w if there is an epistemic model \mathcal{W} of T_w^* , such that $C^{\mathcal{I}_w, \mathcal{W}} \neq \emptyset$, for some

$\mathcal{I} \in \mathcal{W}$. A concept subsumption $C \sqsubseteq D$ is **valid as witnessed by** T_w , denoted by $C \sqsubseteq_w D$, if, for every epistemic model \mathcal{W} of T_w^* , and every $\mathcal{I} \in \mathcal{W}$ $C^{\mathcal{I}_w, \mathcal{W}} \subseteq D^{\mathcal{I}_w, \mathcal{W}}$. An alternative notation for $C \sqsubseteq_w D$ is $T_w^* \models_w C \sqsubseteq D$.

Example 2 (Semantics): We answer the two questions posed against the knowledge base presented in Example 1.

- (i) The first one concerns the satisfiability of the concept expression

$$\exists_1 \text{teaches.Course} \sqcap_1 \neg_1 \mathbf{K}_1 \text{Faculty} \quad (5.2)$$

as witnessed by module T_1 . Note, first, that the concept expression

$$\exists_1 \text{teaches.Course} \sqcap_1 \neg_1 \text{Faculty}$$

is not satisfiable as witnessed by T_1 , since

$$\begin{aligned} & (\exists_1 \text{teaches.Course} \sqcap_1 \neg_1 \text{Faculty})^{\mathcal{I}_1, \mathcal{W}} \\ &= (\exists_1 \text{teaches.Course})^{\mathcal{I}_1, \mathcal{W}} \cap (\neg_1 \text{Faculty})^{\mathcal{I}_1, \mathcal{W}} \\ &\subseteq \text{Faculty}^{\mathcal{I}_1, \mathcal{W}} \cap (\neg_1 \text{Faculty})^{\mathcal{I}_1, \mathcal{W}} \\ &= \emptyset \end{aligned}$$

However, the concept expression (5.2) is satisfiable as witnessed by T_1 . To see this consider a federated epistemic model \mathcal{W} , with $\Delta^1 = \{x, y\}$ and $\Delta^2 = \emptyset$. Let $\mathcal{I} \in \mathcal{W}$ be the model such that $x \in \text{Faculty}^{\mathcal{I}_1, \mathcal{W}}$, $y \in \text{Course}^{\mathcal{I}_1, \mathcal{W}}$ and $\text{teaches}^{\mathcal{I}_1, \mathcal{W}} = \{(x, y)\}$. Moreover, let $\mathcal{J} \in \mathcal{W}$ be the model with $x, y \in \text{Course}^{\mathcal{J}_1, \mathcal{W}}$ and $\text{Faculty}^{\mathcal{J}_1, \mathcal{W}} = \text{teaches}^{\mathcal{J}_1, \mathcal{W}} = \emptyset$. Then, it is not difficult to see that

$$(\exists_1 \text{teaches.Course} \sqcap_1 \neg_1 \mathbf{K}_1 \text{Faculty})^{\mathcal{I}_1, \mathcal{W}} = \{x\} \neq \emptyset.$$

Thus, the concept expression (5.2) is satisfiable as witnessed by T_1 .

(ii) The second one concerns the satisfiability of

$$\exists_2 \text{teaches} . \exists_2 \text{enrolled} . \text{Student} \sqcap_2 \neg_2 \text{Faculty} \quad (5.3)$$

as witnessed by T_2 . Let $(\mathcal{I}, \mathcal{W})$ be an epistemic model of T , such that

$$(\exists_2 \text{teaches} . \exists_2 \text{enrolled} . \text{Student} \sqcap_2 \neg_2 \text{Faculty})^{\mathcal{I}_2, \mathcal{W}} \neq \emptyset.$$

Thus, there exists an $x \in \Delta^2$, such that

1. $x \in (\exists_2 \text{teaches} . \exists_2 \text{enrolled} . \text{Student})^{\mathcal{I}_2, \mathcal{W}}$ and
2. $x \in (\neg_2 \text{Faculty})^{\mathcal{I}_2, \mathcal{W}} = \Delta^2 \setminus r_{12}(\text{Faculty}^{\mathcal{I}_1, \mathcal{W}})$.

Thus, there exist individuals $x', y' \in \Delta^1$, such that $(x', x), (y', y) \in r_{12}$, $(x', y') \in \text{teaches}^{\mathcal{I}_1, \mathcal{W}}$ and, also, $y \in (\exists_2 \text{enrolled} . \text{Student})^{\mathcal{I}_2, \mathcal{W}} \subseteq r_{12}(\text{Course}^{\mathcal{I}_1, \mathcal{W}})$. But then, by 2 above, $x \notin r_{12}(\text{Faculty}^{\mathcal{I}_1, \mathcal{W}})$, whence $x' \notin \text{Faculty}^{\mathcal{I}_1, \mathcal{W}}$ and, hence, $y' \notin \text{Course}^{\mathcal{I}_1, \mathcal{W}}$. Thus, we get $y' \in \text{Faculty}^{\mathcal{I}_1, \mathcal{W}}$. Therefore $y \in r_{12}(\text{Faculty}^{\mathcal{I}_1, \mathcal{W}})$, contradicting $y \in r_{12}(\text{Course}^{\mathcal{I}_1, \mathcal{W}})$ and $\text{Faculty} \sqcap_2 \text{Course} \sqsubseteq \perp$. This shows that the concept expression (5.3) is not satisfiable as witnessed by T_2 .

5.3 A Reduction from F- \mathcal{ALCIK} to \mathcal{ALCIK}

In [97] (see also [98]), a **reduction** \mathfrak{R} from an F- \mathcal{ALCI} KB $\Sigma_d = \{T_i\}$ to an \mathcal{ALCI} KB $\Sigma := \mathfrak{R}(\Sigma_d)$ was obtained. Since this reduction is going to be reused in the present work in order to obtain a reduction from an F- \mathcal{ALCIK} KB to an \mathcal{ALCIK} KB, we review it here. This reduction will also allow us to show that the satisfiability problem in F- \mathcal{ALCIK} is decidable, since by [35], the same problem in \mathcal{ALCIK} is decidable, and to pinpoint the complexity of the problem.

Let Σ_d be an F- \mathcal{ALCI} KB. The signature of $\Sigma = \mathfrak{R}(\Sigma_d)$ is the union of the local signatures of the modules together with a global top \top , a global bottom \perp , local top concepts \top_i , for all

$i \in V$, and, finally, a collection of new role names $\{R_{ij}\}_{(i,j) \in E}$, i.e.,

$$\text{Sig}(\Sigma) = \bigcup_i (\mathcal{C}_i \cup \mathcal{R}_i) \cup \{\top, \perp\} \cup \{\top_i : 1 \leq i \leq n\} \cup \{R_{ij} : (i, j) \in E\}. \quad (5.4)$$

Moreover, various axioms derived from the structure of Σ_d are added to Σ .

- For each $C \in \mathcal{C}_i$, $C \sqsubseteq \top_i$ is added to Σ .
- For each $R \in \mathcal{R}_i$, \top_i is stipulated to be the domain and range of R , i.e., $\top \sqsubseteq \forall R^-. \top_i$ and $\top \sqsubseteq \forall R. \top_i$ are added to Σ .
- For each new role name R_{ij} , \top_i is stipulated to be its domain and \top_j to be its range, i.e., $\top \sqsubseteq \forall R_{ij}^-. \top_i$ and $\top \sqsubseteq \forall R_{ij}. \top_j$ are added to Σ .
- For each $C \sqsubseteq D \in T_i$, $\#_i(C) \sqsubseteq \#_i(D)$ is added to Σ , where $\#_i$ is a function from $\widehat{\mathcal{C}}_i$ to the set of \mathcal{ALCI} -concepts. The precise definition of $\#_i$ is given below.

The mapping $\#_i(C)$ serves to maintain the compatibility of the concept domains. It is defined by induction on the structure of $C \in \widehat{\mathcal{C}}_i$:

- $\#_i(C) = C$, if $C \in \mathcal{C}_i$;
- $\#_i(C) = \exists R_{ji}^-. \#_j(C)$, if $C \in \mathcal{C}_j \cap \widehat{\mathcal{C}}_i$;
- $\#_i(\neg_j D) = \exists R_{ji}^-. (\neg \#_j(D) \sqcap \top_j)$;
- $\#_i(D \boxplus_j E) = \exists R_{ji}^-. (\#_j(D) \boxplus \#_j(E))$, for $\boxplus = \sqcap$ or $\boxplus = \sqcup$;
- $\#_i(\exists_j R. D) = \exists R_{ji}^-. (\exists R_{kj}^-. (\exists R. (\exists R_{kj}. \#_j(D))))$, for $R \in \mathcal{R}_k \cup \mathcal{R}_k^-$;
- $\#_i(\forall_j R. D) = \exists R_{ji}^-. (\forall R_{kj}^-. (\forall R. (\forall R_{kj}. \#_j(D))))$, for $R \in \mathcal{R}_k \cup \mathcal{R}_k^-$.

The reduction \mathfrak{R} was shown in [97] to be sound and complete in the sense that, if the local top concept \top_w in $\mathfrak{R}(\Sigma_d)$, that corresponds to a module T_w in Σ_d , is satisfiable in an \mathcal{ALCI} -model of $\mathfrak{R}(\Sigma_d)$, then Σ_d itself is consistent as witnessed by T_w and vice-versa.

Suppose, now, that we start from a F- \mathcal{ALCIK} KB Σ_d . We extend the reduction \mathfrak{R} to cover the contextualized epistemic operators. Thus, we obtain a reduction from the F- \mathcal{ALCIK}

KB Σ_d to an \mathcal{ALCIK} KB $\Sigma = \mathfrak{R}(\Sigma_d)$. The signature of Σ is as in Equation (5.4) and the axioms remain the same. The only definition that needs to be expanded to accommodate the epistemic operators is that of the function $\#_i(C)$. More specifically, the set of clauses above is modified and augmented as follows: We first define two auxiliary operators $\#_i^\exists$ and $\#_i^\forall$ on i -role expressions.

- $\#_i^\exists(R) = \exists R_{ki}^- \exists R \exists R_{ki}$, for all $R \in \mathcal{R}_k \cup \mathcal{R}_k^-$;
- $\#_i^\forall(R) = \forall R_{ki}^- \forall R \forall R_{ki}$, for all $R \in \mathcal{R}_k \cup \mathcal{R}_k^-$;
- $\#_i^\exists(\mathbf{K}_j R) = \exists R_{ji}^- \exists \mathbf{K} R \exists R_{ji}$, for all $R \in \mathcal{R}_j \cup \mathcal{R}_j^-$;
- $\#_i^\forall(\mathbf{K}_j R) = \forall R_{ji}^- \forall \mathbf{K} R \forall R_{ji}$, for all $R \in \mathcal{R}_j \cup \mathcal{R}_j^-$;

Then, using the $\#_i^\exists$ and $\#_i^\forall$ -operators, we extend the $\#_i$ -operators defined previously for concept expressions in $\widehat{\mathcal{C}}_i$ to concept expressions in $\widetilde{\mathcal{C}}_i$ by induction on the structure of $C \in \widetilde{\mathcal{C}}_i$, as follows:

- $\#_i(C) = C$, if $C \in \mathcal{C}_i$;
- $\#_i(C) = \exists R_{ji}^- \cdot \#_j(C)$, if $C \in \mathcal{C}_j \cap \widehat{\mathcal{C}}_i$;
- $\#_i(\neg_j D) = \exists R_{ji}^- \cdot (\neg \#_j(D) \sqcap \top_j)$;
- $\#_i(D \boxplus_j E) = \exists R_{ji}^- \cdot (\#_j(D) \boxplus \#_j(E))$, for $\boxplus = \sqcap$ or $\boxplus = \sqcup$;
- $\#_i(\exists_j R.D) = \exists R_{ji}^- \cdot \#_j^\exists(R) \cdot \#_j(D)$, if $R \in \widetilde{\mathcal{R}}_i$;
- $\#_i(\forall_j R.D) = \exists R_{ji}^- \cdot \#_j^\forall(R) \cdot \#_j(D)$, if $R \in \widetilde{\mathcal{R}}_i$;
- $\#_i(\mathbf{K}_j D) = \exists R_{ji}^- \cdot \mathbf{K} \#_j(D)$.

Note how the newly introduced notation enables us to handle uniformly in an elegant way the more complex epistemic roles when it comes to the rules dealing with the existential and the universal role quantifications. When only non-epistemic roles are used, the clauses in the definition collapse to the ones used previously for concept expressions in $\widehat{\mathcal{C}}_i$.

Example 3 (Reducing a Federated Knowledge Base): Let us revisit the F- \mathcal{ALCIK} knowledge base T consisting of two modules T_1 and T_2 , that was presented in Example 1. We provide below the \mathcal{ALCIK} knowledge base $\mathfrak{R}(T)$, that results from T by applying the reduction that was presented in this section. We use horizontal lines to separate the different categories of translated sentences, as were presented in the definition of \mathfrak{R} .

$\text{Faculty} \sqsubseteq \top_1$ $\text{Course} \sqsubseteq \top_1$ $\text{Student} \sqsubseteq \top_2$
$\top \sqsubseteq \forall \text{teaches}^-. \top_1 \quad \top \sqsubseteq \forall \text{teaches}. \top_1$ $\top \sqsubseteq \forall \text{enrolled}^-. \top_2 \quad \top \sqsubseteq \forall \text{enrolled}. \top_2$
$\top \sqsubseteq \forall R_{12}^-. \top_1 \quad \top \sqsubseteq \forall R_{12}. \top_2$
$\top_1 \sqsubseteq \text{Faculty} \sqcup \text{Course}$ $\text{Faculty} \sqcap \text{Course} \sqsubseteq \perp$ $\exists \text{teaches}. \text{Course} \sqsubseteq \text{Faculty}$ $\top_2 \sqsubseteq \exists R_{12}^-. \text{Faculty} \sqcup \exists R_{12}^-. \text{Course} \sqcup \text{Student}$ $\exists R_{12}^-. \text{Faculty} \sqcap \exists R_{12}^-. \text{Course} \sqsubseteq \perp$ $\exists R_{12}^-. \text{Faculty} \sqcap \text{Student} \sqsubseteq \perp$ $\exists R_{12}^-. \text{Course} \sqcap \text{Student} \sqsubseteq \perp$ $\exists \text{enrolled}. \text{Student} \sqsubseteq \exists R_{12}^-. \text{Course}$

We will show in the next section that the extended reduction \mathfrak{R} is sound and complete in the sense that, given a F- \mathcal{ALCIK} KB $\Sigma_d = \{T_i\}_{i \in V}$, if the local top concept \top_w in $\mathfrak{R}(\Sigma_d)$ is satisfiable in an epistemic model $\mathcal{W} = \{\mathcal{I}\}$, satisfying the property

$$\begin{cases} \top_i^{\mathcal{I}} = \top_i^{\mathcal{J}}, \text{ for all } i \in V \text{ and all } \mathcal{I}, \mathcal{J} \in \mathcal{W}, \\ R_{ij}^{\mathcal{I}} = R_{ij}^{\mathcal{J}}, \text{ for all } (i, j) \in E \text{ and all } \mathcal{I}, \mathcal{J} \in \mathcal{W}. \end{cases} \quad (5.5)$$

then Σ_d itself is consistent as witnessed by T_w and vice-versa.

5.4 Soundness and Completeness of the Reduction \mathfrak{R}

Let us call an epistemic model of $\mathfrak{R}(\Sigma_d)$ satisfying Condition (5.5) a **coordinated model**. The strategy that we will follow in order to show that the local top concept \top_w in $\mathfrak{R}(\Sigma_d)$ is satisfiable in a coordinated model iff Σ_d is consistent as witnessed by T_w can be summarized as follows: First, given an epistemic coordinated \mathcal{ALCITK} model \mathcal{W} of the KB $\mathfrak{R}(\Sigma_d)$, we will construct an epistemic F- \mathcal{ALCITK} model $F(\mathcal{W})$ of Σ_d . Then, we will show that if \top_w is satisfiable in the model \mathcal{W} , then $F(\mathcal{W})$ is a model of T_w^* , such that $\Delta^w \neq \emptyset$. Moreover, given an epistemic F- \mathcal{ALCITK} model \mathcal{W} of the KB Σ_d , we will construct an epistemic coordinated \mathcal{ALCITK} model $G(\mathcal{W})$ of $\mathfrak{R}(\Sigma_d)$ and show that if \mathcal{W} is a model of T_w^* , such that $\Delta^w \neq \emptyset$, then \top_w is satisfiable in the model $G(\mathcal{W})$.

First, we provide the constructions of $F(\mathcal{W})$ and $G(\mathcal{W})$. We urge the reader to keep in mind that, in the first case, the model \mathcal{W} refers to an epistemic model of the \mathcal{ALCITK} KB $\mathfrak{R}(\Sigma_d)$, i.e., \mathcal{W} is a maximal set of \mathcal{ALCITK} interpretations, such that, for all $\mathcal{I} \in \mathcal{W}$, the \mathcal{ALCITK} interpretation $(\mathcal{I}, \mathcal{W})$ is a model of $\mathfrak{R}(\Sigma_d)$, in the sense of [35], satisfying the extra Condition (5.5). On the other hand, in $G(\mathcal{W})$, the model \mathcal{W} refers to an epistemic model of the F- \mathcal{ALCITK} KB Σ_d , i.e., \mathcal{W} is a maximal set of F- \mathcal{ALCITK} interpretations, such that, for all $\mathcal{I} \in \mathcal{W}$, the F- \mathcal{ALCITK} interpretation $(\mathcal{I}, \mathcal{W})$ is a model of Σ_d . Hopefully, this mild overloading of notation will not cause any confusion.

Definition 70 *Let $\Sigma_d = \{T_i\}_{i \in V}$ be an F- \mathcal{ALCITK} KB and $(\mathcal{I}, \mathcal{W})$ an interpretation of the \mathcal{ALCITK} KB $\mathfrak{R}(\Sigma_d)$, satisfying Condition (5.5). For each $\mathcal{J} \in \mathcal{W} \cup \{\mathcal{I}\}$, construct an interpretation $F(\mathcal{J}) = \langle \{\mathcal{J}_i\}_{i \in V}, \{r_{ij}\}_{(i,j) \in E} \rangle$ for Σ_d as follows:*

- $\Delta^{\mathcal{J}_i} = \top_i^{\mathcal{J}}$, for all $i \in V$;
- $C^{\mathcal{J}_i} = C^{\mathcal{J}}$, for every $C \in \mathcal{C}_i$;
- $R^{\mathcal{J}_i} = R^{\mathcal{J}}$, for every $R \in \mathcal{R}_i$;
- $r_{ij} = R_{ij}^{\mathcal{J}}$, for every $(i, j) \in E$.

Let $F(\mathcal{W}) = \{F(\mathcal{J}) : \mathcal{J} \in \mathcal{W}\}$ and $F(\mathcal{I}, \mathcal{W}) = (F(\mathcal{I}), F(\mathcal{W}))$.

Note how Condition (5.5) is necessary so that the resulting F- \mathcal{ALCIK} interpretation satisfy the requirements of Definition 69.

Definition 71 Suppose that Σ_d is an F- \mathcal{ALCIK} KB and that $(\mathcal{I}, \mathcal{W})$, with $\mathcal{J} = \langle \{\mathcal{J}_i\}_{i \in V}, \{r_{ij}\}_{(i,j) \in E} \rangle$, for all $\mathcal{J} \in \mathcal{W} \cup \{\mathcal{I}\}$, is an interpretation for Σ_d . For every $\mathcal{J} \in \mathcal{W} \cup \{\mathcal{I}\}$, construct an interpretation $\mathbf{G}(\mathcal{J}) = \langle \Delta^{\mathbf{G}(\mathcal{J})}, \cdot^{\mathbf{G}(\mathcal{J})} \rangle$ of $\mathfrak{R}(\Sigma_d)$ as follows:

- $\Delta^{\mathbf{G}(\mathcal{J})} = \bigcup_{i \in V} \Delta^{\mathcal{J}_i}$;
- $\top_i^{\mathbf{G}(\mathcal{J})} = \Delta^{\mathcal{J}_i}$, for every $i \in V$;
- $C^{\mathbf{G}(\mathcal{J})} = C^{\mathcal{J}_i}$, for every $C \in \mathcal{C}_i$;
- $R^{\mathbf{G}(\mathcal{J})} = R^{\mathcal{J}_i}$, for every $R \in \mathcal{R}_i$;
- $R_{ij}^{\mathbf{G}(\mathcal{J})} = r_{ij}$, for every $(i, j) \in E$.

Furthermore, define $\mathbf{G}(\mathcal{W}) = \{\mathbf{G}(\mathcal{J}) : \mathcal{J} \in \mathcal{W}\}$ and $\mathbf{G}(\mathcal{I}, \mathcal{W}) = (\mathbf{G}(\mathcal{I}), \mathbf{G}(\mathcal{W}))$.

The following technical lemma that was proven for the case of a F- \mathcal{ALCI} KB Σ_d in [97] (see [98] for proof) is extended here to cover the case of an F- \mathcal{ALCIK} KB Σ_d . It will aid in proving the main soundness and completeness results of this section.

Lemma 72 Let Σ_d be an F- \mathcal{ALCIK} KB and $(\mathcal{I}, \mathcal{W})$ an interpretation for $\mathfrak{R}(\Sigma_d)$ satisfying Condition (5.5). For every concept expression $C \in \tilde{\mathcal{C}}_i$, such that $\exists R_{ij}^-.C$ occurs in $\mathfrak{R}(\Sigma_d)$, we have $R_{ij}^{\mathcal{I}, \mathcal{W}}(C^{\mathcal{I}, \mathcal{W}}) = (\exists R_{ij}^-.C)^{\mathcal{I}, \mathcal{W}}$.

Proof:

The proof can be carried out almost verbatim as that of Lemma 10 of [98]. We omit the details. ■

To connect the interpretation $(\mathcal{I}, \mathcal{W})$ with its federated counterpart $(\mathbf{F}(\mathcal{I}), \mathbf{F}(\mathcal{W}))$, we need to establish a correspondence between the interpretation of the translation $\#_i(C)$ of a concept $C \in \tilde{\mathcal{C}}_i$ under $(\mathcal{I}, \mathcal{W})$ and that of the concept C under $(\mathbf{F}(\mathcal{I}), \mathbf{F}(\mathcal{W}))$. This relationship is explored in Lemma 75. For the proof of the lemma, we need to know the exact relationship

between the interpretation of a translation of an existential sentence using a complex role expression in a model $(\mathcal{I}, \mathcal{W})$ of $\mathfrak{R}(\Sigma_d)$ and the interpretation of the sentence itself in $\mathbf{F}(\mathcal{I}, \mathcal{W})$. More precisely, we have

Lemma 73 *Let Σ_d be an F -ALC \mathcal{IK} KB, $(\mathcal{I}, \mathcal{W})$ an interpretation for $\mathfrak{R}(\Sigma_d)$, satisfying Condition (5.5), and $\exists_i R.D \in \tilde{\mathcal{C}}_i$, such that $\#_i(D)^{\mathcal{I}, \mathcal{W}} = D^{\mathbf{F}(\mathcal{I})_i, \mathbf{F}(\mathcal{W})}$. Then*

$$(\#_i^{\exists}(R).\#_i(D))^{\mathcal{I}, \mathcal{W}} = \{x \in \Delta^i : (\exists y \in D^{\mathbf{F}(\mathcal{I})_i, \mathbf{F}(\mathcal{W})})((x, y) \in R^{\mathbf{F}(\mathcal{I})_i, \mathbf{F}(\mathcal{W})})\}.$$

Proof:

Assume, first, that $R \in \mathcal{R}_k \cup \mathcal{R}_k^-$. Then, we have

$$\begin{aligned} & (\#_i^{\exists}(R).\#_i(D))^{\mathcal{I}, \mathcal{W}} \\ &= (\exists R_{ki}^-. \exists R. \exists R_{ki}. \#_i(D))^{\mathcal{I}, \mathcal{W}} \\ &= \{x \in \Delta^{\mathcal{I}, \mathcal{W}} : (\exists y \in \#_i(D)^{\mathcal{I}, \mathcal{W}})((x, y) \in R_{ki}^{\mathcal{I}, \mathcal{W}}(R^{\mathcal{I}, \mathcal{W}}))\} \\ &= \{x \in \Delta^i : (\exists y \in D^{\mathbf{F}(\mathcal{I})_i, \mathbf{F}(\mathcal{W})})((x, y) \in r_{ki}(R^{\mathbf{F}(\mathcal{I})_k, \mathbf{F}(\mathcal{W})}))\} \\ &= \{x \in \Delta^i : (\exists y \in D^{\mathbf{F}(\mathcal{I})_i, \mathbf{F}(\mathcal{W})})((x, y) \in R^{\mathbf{F}(\mathcal{I})_i, \mathbf{F}(\mathcal{W})})\}. \end{aligned}$$

If, on the other hand, $R \in \mathcal{R}_j \cup \mathcal{R}_j^-$, we get

$$\begin{aligned} & (\#_i^{\exists}(\mathbf{K}_j R).\#_i(D))^{\mathcal{I}, \mathcal{W}} \\ &= (\exists R_{ji}^-. \exists \mathbf{K} R. \exists R_{ji}. \#_i(D))^{\mathcal{I}, \mathcal{W}} \\ &= \{x \in \Delta^{\mathcal{I}, \mathcal{W}} : (\exists y \in \#_i(D)^{\mathcal{I}, \mathcal{W}})((x, y) \in R_{ji}^{\mathcal{I}, \mathcal{W}}((\mathbf{K} R)^{\mathcal{I}, \mathcal{W}}))\} \\ &= \{x \in \Delta^{\mathcal{I}, \mathcal{W}} : (\exists y \in \#_i(D)^{\mathcal{I}, \mathcal{W}})((x, y) \in R_{ji}^{\mathcal{I}, \mathcal{W}}(\bigcap_{\mathcal{J} \in \mathcal{W}} R^{\mathcal{J}, \mathcal{W}}))\} \\ &= \{x \in \Delta^i : (\exists y \in D^{\mathbf{F}(\mathcal{I})_i, \mathbf{F}(\mathcal{W})})((x, y) \in r_{ji}(\bigcap_{\mathcal{J} \in \mathcal{W}} R^{\mathbf{F}(\mathcal{J})_j, \mathbf{F}(\mathcal{W})}))\} \\ &= \{x \in \Delta^i : (\exists y \in D^{\mathbf{F}(\mathcal{I})_i, \mathbf{F}(\mathcal{W})})((x, y) \in (\mathbf{K}_j R)^{\mathbf{F}(\mathcal{I})_i, \mathbf{F}(\mathcal{W})})\}. \end{aligned}$$

■

A similar result holds for universal role quantifications. It will also help us in proving Lemma 75, which establishes the main connection between extensions in a coordinated model

$(\mathcal{I}, \mathcal{W})$ of $\mathfrak{R}(\Sigma_d)$ of translated sentences and corresponding extensions in $\mathbf{F}(\mathcal{I}, \mathcal{W})$ of the original sentences.

Lemma 74 *Let Σ_d be an F - \mathcal{ALCIK} KB, $(\mathcal{I}, \mathcal{W})$ be an interpretation for $\mathfrak{R}(\Sigma_d)$, satisfying Condition (5.5), and $\forall_i R.D \in \tilde{\mathcal{C}}_i$ such that $\#_i(D)^{\mathcal{I}, \mathcal{W}} = D^{\mathbf{F}(\mathcal{I})_i, \mathbf{F}(\mathcal{W})}$. Then*

$$\begin{aligned} & (\#_i^{\forall}(R). \#_i(D))^{\mathcal{I}, \mathcal{W}} \\ &= \{x \in \Delta^i : (\forall y \in \Delta^i)((x, y) \in R^{\mathbf{F}(\mathcal{I})_i, \mathbf{F}(\mathcal{W})} \text{ implies } y \in D^{\mathbf{F}(\mathcal{I})_i, \mathbf{F}(\mathcal{W})})\}. \end{aligned}$$

Proof:

Assume, first, that $R \in \mathcal{R}_k \cup \mathcal{R}_k^-$. Then, we have

$$\begin{aligned} & (\#_i^{\exists}(R). \#_i(D))^{\mathcal{I}, \mathcal{W}} \\ &= (\forall R_{ki}^-. \forall R. \forall R_{ki}. \#_i(D))^{\mathcal{I}, \mathcal{W}} \\ &= \{x \in \Delta^{\mathcal{I}, \mathcal{W}} : (\forall y \in \Delta^{\mathcal{I}, \mathcal{W}})((x, y) \in R_{ki}^{\mathcal{I}, \mathcal{W}}(R^{\mathcal{I}, \mathcal{W}}) \\ & \hspace{15em} \text{implies } y \in \#_i(D)^{\mathcal{I}, \mathcal{W}})\} \\ &= \{x \in \Delta^i : (\forall y \in \Delta^i)((x, y) \in r_{ki}(R^{\mathbf{F}(\mathcal{I})_k, \mathbf{F}(\mathcal{W})}) \\ & \hspace{15em} \text{implies } y \in D^{\mathbf{F}(\mathcal{I})_i, \mathbf{F}(\mathcal{W})})\} \\ &= \{x \in \Delta^i : (\forall y \in \Delta^i)((x, y) \in R^{\mathbf{F}(\mathcal{I})_i, \mathbf{F}(\mathcal{W})} \text{ implies } y \in D^{\mathbf{F}(\mathcal{I})_i, \mathbf{F}(\mathcal{W})})\}. \end{aligned}$$

If, on the other hand, $R \in \mathcal{R}_j \cup \mathcal{R}_j^-$, we get

$$(\#_i^{\forall}(\mathbf{K}_j R). \#_i(D))^{\mathcal{I}, \mathcal{W}}$$

$$\begin{aligned}
&= (\forall R_{ji}^-. \forall \mathbf{K}R. \forall R_{ji}. \#_i(D))^{\mathcal{I}, \mathcal{W}} \\
&= \{x \in \Delta^{\mathcal{I}, \mathcal{W}} : (\forall y \in \Delta^{\mathcal{I}, \mathcal{W}})((x, y) \in R_{ji}^{\mathcal{I}, \mathcal{W}}((\mathbf{K}R)^{\mathcal{I}, \mathcal{W}})) \\
&\quad \text{implies } y \in \#_i(D)^{\mathcal{I}, \mathcal{W}})\} \\
&= \{x \in \Delta^{\mathcal{I}, \mathcal{W}} : (\forall y \in \Delta^{\mathcal{I}, \mathcal{W}})((x, y) \in R_{ji}^{\mathcal{I}, \mathcal{W}}(\bigcap_{\mathcal{J} \in \mathcal{W}} R^{\mathcal{J}, \mathcal{W}})) \\
&\quad \text{implies } y \in \#_i(D)^{\mathcal{I}, \mathcal{W}})\} \\
&= \{x \in \Delta^i : (\forall y \in \Delta^i)((x, y) \in r_{ji}(\bigcap_{\mathcal{J} \in \mathcal{W}} R^{\mathbf{F}(\mathcal{J})_j, \mathbf{F}(\mathcal{W})})) \\
&\quad \text{implies } y \in D^{\mathbf{F}(\mathcal{I})_i, \mathbf{F}(\mathcal{W})})\} \\
&= \{x \in \Delta^i : (\forall y \in \Delta^i)((x, y) \in (\mathbf{K}_j R)^{\mathbf{F}(\mathcal{I})_i, \mathbf{F}(\mathcal{W})}) \\
&\quad \text{implies } y \in D^{\mathbf{F}(\mathcal{I})_i, \mathbf{F}(\mathcal{W})})\}.
\end{aligned}$$

■

Lemma 75 Suppose that Σ_d is an *F-ALCICK* KB, $(\mathcal{I}, \mathcal{W})$ an interpretation for $\mathfrak{R}(\Sigma_d)$, satisfying Condition (5.5), and $\mathbf{F}(\mathcal{I}, \mathcal{W})$, with $\mathcal{J} = \langle \{\mathcal{J}_i\}_{i \in V}, \{r_{ij}\}_{(i,j) \in E} \rangle$, for all $\mathcal{J} \in \mathbf{F}(\mathcal{W}) \cup \{\mathbf{F}(\mathcal{I})\}$, such that $\mathcal{J}_i = \langle \Delta^{\mathcal{J}_i}, \cdot^{\mathcal{J}_i} \rangle, i \in V$. Then

$$\#_i(C)^{\mathcal{I}, \mathcal{W}} = C^{\mathbf{F}(\mathcal{I})_i, \mathbf{F}(\mathcal{W})}, \quad \text{for every } C \in \tilde{\mathcal{C}}_i, \quad i \in V.$$

Proof:

We do this by structural induction on C . All steps of the induction that involve operators other than either the epistemic role or the epistemic concept operators can be carried out exactly as in the proof of Lemma 12 of [98]. Therefore, we will demonstrate in detail only the case where the main (outermost) connective is an existential, universal or epistemic concept operator.

Suppose that $C = \exists_j R.D$, with $R \in \tilde{\mathcal{R}}_i$. Then

$$\begin{aligned}
\#_i(\exists_j R.D)^{\mathcal{I}, \mathcal{W}} &= (\exists R_{ji}^-. \#_j^\exists R. \#_j(D))^{\mathcal{I}, \mathcal{W}} \\
&\quad (\text{by the definition of } \#_i(\exists_j R.D)) \\
&= R_{ji}^{\mathcal{I}, \mathcal{W}}((\#_j^\exists R. \#_j(D))^{\mathcal{I}, \mathcal{W}}) \quad (\text{by Lemma 72}) \\
&= r_{ji}(\{x \in \Delta^j : (\exists y \in D^{\mathbf{F}(\mathcal{I})_j, \mathbf{F}(\mathcal{W})})(x, y) \in \\
&\quad R^{\mathbf{F}(\mathcal{I})_j, \mathbf{F}(\mathcal{W})}\}) \\
&\quad (\text{by Lemma 73 and the induction hypothesis}) \\
&= r_{ji}((\exists_j R.D)^{\mathbf{F}(\mathcal{I})_j, \mathbf{F}(\mathcal{W})}) \\
&= (\exists_j R.D)^{\mathbf{F}(\mathcal{I})_i, \mathbf{F}(\mathcal{W})}.
\end{aligned}$$

The case of a universal role quantification may be handled similarly. The only difference is that Lemma 74 should be used in place of Lemma 73.

Suppose, finally, that $C = \mathbf{K}_j D$. Then

$$\begin{aligned}
\#_i(\mathbf{K}_j D)^{\mathcal{I}, \mathcal{W}} &= (\exists R_{ji}^-. \mathbf{K} \#_j(D))^{\mathcal{I}, \mathcal{W}} \quad (\text{by the definition of } \#_i(\mathbf{K}_j D)) \\
&= R_{ji}^{\mathcal{I}, \mathcal{W}}((\mathbf{K} \#_j(D))^{\mathcal{I}, \mathcal{W}}) \quad (\text{by Lemma 72}) \\
&= R_{ji}^{\mathcal{I}, \mathcal{W}}(\bigcap_{\mathcal{J} \in \mathcal{W}} \#_j(D)^{\mathcal{J}, \mathcal{W}}) \quad (\text{by the definition of } \cdot^{\mathcal{I}, \mathcal{W}}) \\
&= r_{ji}(\bigcap_{\mathcal{J} \in \mathcal{W}} D^{\mathbf{F}(\mathcal{J})_i, \mathbf{F}(\mathcal{W})}) \quad (\text{by the definition of } \mathbf{F}(\mathcal{I}, \mathcal{W}) \\
&\quad \text{and the induction hypothesis}) \\
&= (\mathbf{K}_j D)^{\mathbf{F}(\mathcal{I})_i, \mathbf{F}(\mathcal{W})}. \quad (\text{by the definition of } (\mathbf{K}_j D)^{\mathbf{F}(\mathcal{I})_i, \mathbf{F}(\mathcal{W})})
\end{aligned}$$

■

The following is our main soundness theorem for the reduction \mathfrak{R} .

Theorem 76 (Soundness) *Let Σ_d be an $F\text{-}\mathcal{ALCIK}$ KB, and T_w a module of Σ_d . If \top_w is satisfiable in a coordinated model of $\mathfrak{R}(T_w^*)$, then Σ_d is consistent as witnessed by T_w .*

Proof:

Suppose that \top_w is satisfiable in a coordinated model of $\mathfrak{R}(T_w^*)$. Assume $(\mathcal{I}, \mathcal{W})$, with $\mathcal{I} = \langle \Delta^{\mathcal{I}}, \cdot^{\mathcal{I}} \rangle$, such that $\top_w^{\mathcal{I}} \neq \emptyset$ is a coordinated model of $\mathfrak{R}(T_w^*)$. Our goal is to show that

$(\mathbf{F}(\mathcal{I}), \mathbf{F}(\mathcal{W}))$ is an epistemic model of T_w^* , such that $\Delta^w := \Delta^{\mathbf{F}(\mathcal{I})_w, \mathbf{F}(\mathcal{W})} \neq \emptyset$.

Clearly, we have $\Delta^w = \top_w^{\mathcal{I}, \mathcal{W}} \neq \emptyset$, by the hypothesis. So it suffices to show that $(\mathbf{F}(\mathcal{I}), \mathbf{F}(\mathcal{W}))$ is a model of the federated ontology T_w^* , i.e., that it satisfies $(\mathbf{F}(\mathcal{I}), \mathbf{F}(\mathcal{W})) \models_i T_i$, for every $i \in V_w$. Suppose that $C \sqsubseteq D \in T_i$. By the construction of $\mathfrak{R}(T_w^*)$ and the fact that $(\mathcal{I}, \mathcal{W}) \models \mathfrak{R}(T_w^*)$, we must have $\#_i(C)^{\mathcal{I}, \mathcal{W}} \subseteq \#_i(D)^{\mathcal{I}, \mathcal{W}}$, whence, by Lemma 75, we obtain that $C^{\mathbf{F}(\mathcal{I})_i, \mathbf{F}(\mathcal{W})} \subseteq D^{\mathbf{F}(\mathcal{I})_i, \mathbf{F}(\mathcal{W})}$, showing that $(\mathbf{F}(\mathcal{I}), \mathbf{F}(\mathcal{W})) \models T_w^*$. ■

To connect the federated interpretation $(\mathcal{I}, \mathcal{W})$ with its single module-counterpart $\mathbf{G}(\mathcal{I}, \mathcal{W})$, we need to establish a correspondence between the interpretation of the translation $\#_i(C)$ of a concept $C \in \tilde{\mathcal{C}}_i$ under $\mathbf{F}(\mathcal{I}, \mathcal{W})$ and that of the concept C under $(\mathcal{I}, \mathcal{W})$.

Lemma 77 *Let Σ_d be an F-ALCCK KB and $(\mathcal{I}, \mathcal{W})$, with $\mathcal{W} = \{\mathcal{J}\}$ and $\mathcal{J} = \langle \{\mathcal{J}_i\}_{i \in V}, \{r_{ij}\}_{(i,j) \in E} \rangle$, an epistemic model of Σ_d . Then*

$$\#_i(C)^{\mathbf{G}(\mathcal{I}), \mathbf{G}(\mathcal{W})} = C^{\mathcal{I}, \mathcal{W}}, \text{ for every } C \in \tilde{\mathcal{C}}_i, i \in V.$$

Proof:

As in the proof of Lemma 17 of [98], one may show that, for every $\mathcal{J} \in \mathcal{W}$, we have that $\mathcal{J} = \mathbf{F}(\mathbf{G}(\mathcal{J}))$. Therefore, by Lemma 75, $\#_i(C)^{\mathbf{G}(\mathcal{I}), \mathbf{G}(\mathcal{W})} = C^{\mathbf{F}(\mathbf{G}(\mathcal{I}))_i, \mathbf{F}(\mathbf{G}(\mathcal{W}))} = C^{\mathcal{I}, \mathcal{W}}$. ■

Having at hand Lemma 77, we can now show that the converse of Theorem 76 also holds. That is, if an F-ALCCK ontology Σ_d is consistent as witnessed by a specific module T_w , then the corresponding local top \top_w is satisfiable in a coordinated model of $\mathfrak{R}(T_w^*)$. More precisely, we have the following

Theorem 78 (Completeness) *Let $\Sigma_d = \{T_i\}_{i \in V}$ be an F-ALCCK ontology. If Σ_d is consistent as witnessed by a module T_w , then \top_w is satisfiable in a coordinated model of $\mathfrak{R}(T_w^*)$.*

Proof:

Suppose that Σ_d is consistent as witnessed by T_w . Thus, it has an epistemic model $(\mathcal{I}, \mathcal{W})$, with $\mathcal{J} = \langle \{\mathcal{J}_i\}_{i \in V}, \{r_{ij}\}_{(i,j) \in E} \rangle$, $\mathcal{J} \in \mathcal{W} \cup \{\mathcal{I}\}$, satisfying $\Delta^{\mathcal{I}, \mathcal{W}} \neq \emptyset$. We show that $(\mathbf{G}(\mathcal{I}), \mathbf{G}(\mathcal{W}))$ is a model of $\mathfrak{R}(T_w^*)$, such that $\top_w^{\mathbf{G}(\mathcal{I})} \neq \emptyset$.

We have $\top_w^{\mathcal{G}(\mathcal{I}), \mathcal{G}(\mathcal{W})} = \Delta^{\mathcal{I}_w, \mathcal{W}} \neq \emptyset$, by the hypothesis.

Clearly, if $C \in \mathcal{C}_i$, then $C^{\mathcal{G}(\mathcal{I}), \mathcal{G}(\mathcal{W})} = C^{\mathcal{G}(\mathcal{I})} = C^{\mathcal{I}_i, \mathcal{W}} \subseteq \Delta^i = \top_i^{\mathcal{G}(\mathcal{I}), \mathcal{G}(\mathcal{W})}$, whence $C \sqsubseteq \top_i$ holds in $(\mathcal{G}(\mathcal{I}), \mathcal{G}(\mathcal{W}))$.

Next, suppose that $R \in \mathcal{R}_i$ and let $x \in \Delta^{\mathcal{G}(\mathcal{I}), \mathcal{G}(\mathcal{W})} = \Delta^{\mathcal{G}(\mathcal{I})} = \bigcup_{i \in V} \Delta^i$. Assume that $y \in \Delta^{\mathcal{G}(\mathcal{I})}$, such that $(x, y) \in (R^{\mathcal{G}(\mathcal{I}), \mathcal{G}(\mathcal{W})})^- = (R^{\mathcal{G}(\mathcal{I})})^-$, i.e., $(y, x) \in R^{\mathcal{G}(\mathcal{I})} = R^{\mathcal{I}_i, \mathcal{W}}$. Thus, we must have $y \in \Delta^i = \top_i^{\mathcal{G}(\mathcal{I}), \mathcal{G}(\mathcal{W})}$, whence $x \in \{t \in \Delta^{\mathcal{G}(\mathcal{I}), \mathcal{G}(\mathcal{W})} : (\forall y \in \Delta^{\mathcal{G}(\mathcal{I}), \mathcal{G}(\mathcal{W})})((t, y) \in R^{\mathcal{G}(\mathcal{I}), \mathcal{G}(\mathcal{W})} \rightarrow y \in \top_i^{\mathcal{G}(\mathcal{I}), \mathcal{G}(\mathcal{W})})\} = (\forall R^-. \top_i)^{\mathcal{G}(\mathcal{I}), \mathcal{G}(\mathcal{W})}$. This shows that $\top \sqsubseteq \forall R^-. \top_i$ also holds in $(\mathcal{G}(\mathcal{I}), \mathcal{G}(\mathcal{W}))$. The fact that $(\mathcal{G}(\mathcal{I}), \mathcal{G}(\mathcal{W})) \models \top \sqsubseteq \forall R. \top_i$ may be shown similarly. Also along the same lines follow the proofs that the two concept inclusion axioms $\top \sqsubseteq \forall R_{ij}^-. \top_i$ and $\top \sqsubseteq \forall R_{ij}. \top_j$ are valid in $(\mathcal{G}(\mathcal{I}), \mathcal{G}(\mathcal{W}))$.

Finally, suppose that $\#_i(C) \sqsubseteq \#_i(D)$ is in $\mathfrak{R}(\Sigma_d)$. Then $C \sqsubseteq D \in T_i$ and, since $(\mathcal{I}, \mathcal{W}) \models \Sigma_d$, we must have $C^{\mathcal{I}_i, \mathcal{W}} \subseteq D^{\mathcal{I}_i, \mathcal{W}}$. Therefore, by Lemma 77, $\#_i(C)^{\mathcal{G}(\mathcal{I}), \mathcal{G}(\mathcal{W})} \subseteq \#_i(D)^{\mathcal{G}(\mathcal{I}), \mathcal{G}(\mathcal{W})}$, whence $(\mathcal{G}(\mathcal{I}), \mathcal{G}(\mathcal{W})) \models \#_i(C) \sqsubseteq \#_i(D)$. Thus, if $(\mathcal{I}, \mathcal{W}) \models T_w^*$, we must have that $(\mathcal{G}(\mathcal{I}), \mathcal{G}(\mathcal{W})) \models \mathfrak{R}(T_w^*)$. This concludes the proof that, if Σ_d is consistent as witnessed by a package T_w , then \top_w is satisfiable with respect to $\mathfrak{R}(T_w^*)$. (Note that $(\mathcal{G}(\mathcal{I}), \mathcal{G}(\mathcal{W}))$ is indeed a coordinated model due to Definition 69.) ■

Consequences of Soundness and Completeness

By combining Theorems 76 and 78 we get the following

Theorem 79 (Soundness and Completeness) *Let $\Sigma_d = \{T_i\}_{i \in V}$ be an $F\text{-}\mathcal{ALCIK}$ ontology. Σ_d is consistent as witnessed by a module T_w if and only if \top_w is satisfiable in a coordinated model of $\mathfrak{R}(T_w^*)$.*

It is well known that concept satisfiability in \mathcal{ALCI} with respect to arbitrary TBoxes is EXPTIME-hard [84, 1]. Thus concept satisfiability in \mathcal{ALCIK} is also EXPTIME-hard. By the results presented in [35], it can be shown that concept satisfiability in \mathcal{ALCIK} is in EXPTIME. Therefore, since the reduction \mathfrak{R} can be performed in polynomial time, we obtain

Theorem 80 *Deciding TBox consistency in $F\text{-}\mathcal{ALCIK}$ is EXPTIME-complete.*

The following theorem, which is a consequence of Theorem 79, shows that a given subsumption is valid as witnessed by a module T_i of an F- \mathcal{ALCIK} ontology T if and only if its translation under $\#_i$ is valid in any coordinated model of $\mathfrak{R}(T_i^*)$.

Theorem 81 (Subsumption Preservation) *Let $\Sigma_d = \{T_i\}_{i \in V}$ be an F- \mathcal{ALCIK} ontology. Then $T_i^* \models_i C \sqsubseteq D$ iff $\#_i(C) \sqsubseteq \#_i(D)$ is satisfied in every coordinated model of $\mathfrak{R}(T_i^*)$.*

Proof:

Suppose, first, that $T_i^* \models_i C \sqsubseteq D$ and let $(\mathcal{I}, \mathcal{W})$ be a coordinated epistemic model of $\mathfrak{R}(T_i^*)$. Then, by Theorem 76, $(F(\mathcal{I}), F(\mathcal{W}))$ is a model of T_i^* , whence, since $T_i^* \models_i C \sqsubseteq D$, we get that $(F(\mathcal{I}), F(\mathcal{W})) \models_i C \sqsubseteq D$. This implies that $(\mathcal{I}, \mathcal{W}) \models \#_i(C) \sqsubseteq \#_i(D)$.

Conversely, suppose that every coordinated model of $\mathfrak{R}(T_i^*)$ satisfies $\#_i(C) \sqsubseteq \#_i(D)$ and let $(\mathcal{I}, \mathcal{W}) \models T_i^*$. Then, by Theorem 78, $(G(\mathcal{I}), G(\mathcal{W}))$ is a coordinated model of $\mathfrak{R}(T_i^*)$. Thus, we get that $(G(\mathcal{I}), G(\mathcal{W})) \models \#_i(C) \sqsubseteq \#_i(D)$. This implies that $(\mathcal{I}, \mathcal{W}) \models_i C \sqsubseteq D$. Therefore $T_i^* \models_i C \sqsubseteq D$. ■

5.5 Summary

In this paper we have introduced a modular ontology language, contextualized federated description logic F- \mathcal{ALCIK} , that allows reuse of knowledge from multiple ontologies. An F- \mathcal{ALCIK} ontology consists of multiple ontology modules each of which can be viewed as an \mathcal{ALCIK} ontology and has several important features that, to the best of our knowledge, appear together in a single ontology language for the first time. Namely,

- it is modular, i.e., provides a semantics for independently developed, autonomous but interrelated modules, that can share parts of their vocabulary without the need to redefine their meanings;
- it is fully contextualized, since all its logical connectives assume a different meaning when used in different modules; a module using a “foreign” connective is calling for its extension to be computed in another local domain and then imported via inter-domain relations;

- it provides support for the use of contextualized epistemic operators in each module; whereas the epistemic role operators can be applied only in front of roles defined in their own module, the epistemic concept operators can be applied in front of arbitrary concept expressions, as long as the necessary importing relations are fulfilled.

A case has been made in [35] that the knowledge operators allow non-monotonic reasoning, some forms of closed-world reasoning, formulation of procedural rules as well as various forms of more sophisticated queries. Since our language, when restricted to a single module, has the same semantics as that in [35], these nice properties are inherited in the present context.

F-*ALCIK* supports contextualized interpretations, i.e., interpretations from the point of view of a specific module. We have provided a reduction from an F-*ALCIK* ontology to an *ALCIK* ontology that allowed us to use the results of [35] to show that the consistency problem for a F-*ALCIK* ontology is EXPTIME-complete.

It is an interesting problem to develop a distributed reasoning algorithm for F-*ALCIK* by extending the algorithm for F-*ALCI*, that was presented in [99]. This algorithm was based, in turn, in similar work of Bao et al. [4, 6] and Pan et al. [73].

BIBLIOGRAPHY

- [1] Baader, F., Calvanese, D., McGuinness, D.L., Nardi, D., and Patel-Schneider, P.F., editors, *The Description Logic Handbook: Theory, Implementation, and Applications*, Cambridge University Press, 2003
- [2] Baader, F., and Nutt, W., Basic description logics, In Baader, F., et.al., editors, *The Description Logic Handbook: Theory, Implementation, and Applications*, pp. 43-95, Cambridge University Press, 2003
- [3] Baader, F., and Sattler, U., An Overview of Tableau Algorithms for Description Logics, *Studia Logica*, Vol. 69, No. 1 (2001), pp. 5-40
- [4] Bao, J., Caragea, D., and Honavar, V., A Tableau-Based Federated Reasoning Algorithm for Modular Ontologies, In *IEEE/WIC/ACM International Conference on Web Intelligence*, pp. 404-410. IEEE Press, 2006
- [5] Bao, J., Slutzki, G., and Honavar, V., A Semantic Importing Approach to Knowledge Reuse from Multiple Ontologies, In *AAAI*, pp 1304-1309, 2007
- [6] Bao, J., Slutzki, G., Voutsadakis, G., and Honavar, V., Distributed Reasoning with Modular Ontologies, Technical Report, Iowa State University, 2008
- [7] Bao, J., Voutsadakis, G., Slutzki, G., and Honavar, H., On the Decidability of Role Mappings Between Modular Ontologies, In *AAAI*, pp. 400-05, 2008.
- [8] Bao, J., Voutsadakis, G., Slutzki, G., and Honavar, V., Package-based Description Logics, In H. Stuckenschmidt et al. (eds.), *Lecture Notes in Computer Science*, Vol. 5445 (2009), pp. 349-371, Springer-Verlag, 2009

- [9] Bechhofer, S., van Harmelen, F., Hendler, J., Horrocks, I., McGuinness, D.L., Patel-Schneider, P.F., and Stein, L.A., Owl Web Ontology Language Reference, Technical Report, W3C, 2004
- [10] Benslimane, D., Roussey, C., Vangenot, C., and Arara, A., Towards a Contextual Content of Ontologies, In *ISMIS*, pp. 339-343, 2003
- [11] Berners-Lee, T., Hendler, J., and Lassila, O., The Semantic Web, *Scientific American*, Vol. 284, No. 5 (2001), pp. 34-43
- [12] Bonatti, P.A., and Tettamanzi, A., Some Complexity Results on Fuzzy Description Logics, In Vito Di Gesù, Francesco Masulli, and Alfredo Petrosino, editors, *WILF, Lecture Notes in Computer Science*, Vol. 2955, pp. 19-24, Springer-Verlag, 2003
- [13] Borgida, A., and Serafini, L., Distributed Description Logics: Directed Domain Correspondences in Federated Information Sources, In *CoopIS*, pp. 36-53, 2002
- [14] Borgida, A., and Serafini, L., Distributed Description Logics: Assimilating Information From Peer Sources, *Journal of Data Semantics*, Vol. 1, (2003), pp. 153-184
- [15] Bouquet, P., Giunchiglia, F., van Harmelen, F., Serafini, L., and Stuckenschmidt, H., C-OWL: Contextualizing Ontologies, In Dieter Fensel, Katia P. Sycara, and John Mylopoulos, editors, *International Semantic Web Conference, Lecture Notes in Computer Science*, Vol. 2870, pp. 164-179, Springer-Verlag, 2003
- [16] Brachman, R.J., and Levesque, H.J., The Tractability of Subsumption in Frame-Based Description Languages, In *AAAI*, pp. 34-37, 1984
- [17] Brachman, R.J., Levesque, H.J., and Fikes, R., Krypton: Integrating Terminology and Assertion, In *AAAI*, pp. 31-35, 1983
- [18] Brachman, R.J., and Schmolze, J.G., An Overview of the KL-one Knowledge Representation System, *Cognitive Science*, Vol. 9, No. 2 (1985), pp. 171-216
- [19] Buvac, S., Quantificational logic of context, In *AAAI/IAAI, Vol. 1*, pp. 600-606, 1996

- [20] Buvac, S., Buvac, V., and Mason, I.A., Metamathematics of contexts, *Fundamenta Informatica*, Vol. 23, No. 2/3/4 (1995), pp. 263-301
- [21] Buvac, S., and Kameyama, M., Introduction: Toward a Unified Theory of Context?, *Journal of Logic, Language and Information*, Vol. 7, No. 1 (1998), p. 1
- [22] Buvac, S., and Mason, I.A., Propositional Logic of Context, In *AAAI*, pp. 412-419, 1993
- [23] Cuenca Grau, B., Horrocks, I., Kazakov, Y., and Sattler, U., Modular Reuse of Ontologies: Theory and Practice, *Journal of Artificial Intelligence Research*, Vol. 31 (2008), pp. 273-318
- [24] Cuenca Grau, B., Horrocks, I., Kutz, O., and Sattler, U., Will my Ontologies Fit Together?, In *Proceedings of the 2006 Description Logic Workshop (DL 2006)*, Vol. 189, CEUR (<http://ceur-ws.org/>), 2006
- [25] Cuenca Grau, B., Motik, B., and Kazakov, Y., Import-By-Query: Ontology Reasoning Under Access Limitations, In Craig Boutilier, editor, *IJCAI*, pp. 727-732, 2009
- [26] Cuenca Grau, B., and Parsia, B., From $\mathcal{SHOQ}(\mathbf{D})$ toward \mathcal{E} -connections, In *Description Logics*, 2004
- [27] Cuenca Grau, B., Parsia, B., and Sirin, E., Tableau Algorithms for \mathcal{E} -Connections of Description Logics, Technical Report, University of Maryland Institute for Advanced Computer Studies (UMIACS), TR 2004-72, 2004
- [28] Cuenca Grau, B., Parsia, B., and Sirin, E., Working with Multiple Ontologies on the Semantic Web, In *International Semantic Web Conference*, pp. 620-634, 2004
- [29] Cuenca Grau, B., Parsia, B., and Sirin, E., Combining OWL Ontologies Using \mathcal{E} -Connections, *Journal of Web Semantics*, Vol. 4, No. 1 (2006), pp. 40-59
- [30] Da Costa, P.C.G., *Bayesian semantics for the semantic web*, Ph.D. Dissertation, Fairfax, VA, USA, 2005

- [31] Da Costa, P.C.G., Laskey, K.B., and Laskey, K.J., Pr-OWL: A Bayesian Ontology Language for the Semantic Web, In Da Costa et al. [32], pages 23-33
- [32] Da Costa, P.C.G., Laskey, K.B., Laskey, K.J., and Pool, M., editors. *International Semantic Web Conference, ISWC 2005, Galway, Ireland, Workshop 3: Uncertainty Reasoning for the Semantic Web, 7 November 2005*, 2005
- [33] Ding, Z., and Peng, Y., A Probabilistic Extension to the Web Ontology Language OWL, In *Thirty-Seventh Hawaii International Conference on System Sciences (HICSS-37)*, 2004
- [34] Ding, Z., Peng, Y., and Pan, R., BayesOWL: Uncertainty Modeling in Semantic Web Ontologies, Pages 27, *Studies in Fuzziness and Soft Computing*, Springer-Verlag, October 2005
- [35] Donini, F.M., Lenzerini, M., Nardi, D., Nutt, W., and Schaerf, A., An Epistemic Operator for Description Logics, *Artificial Intelligence*, Vol. 100, No. 1-2 (1998), pp. 225-274
- [36] Fensel, D., Hendler, J., Lieberman, H., and Wahlster, W., Spinning the Semantic Web, In *IEEE Intelligent Systems*, MIT Press, 2003
- [37] Fukushima, Y., Representing Probabilistic Relations in RDF, In Da Costa et al. [32], pp. 106-107
- [38] Ghidini, C., and Giunchiglia, F., Local Models Semantics, or Contextual Reasoning=Locality+Compatibility, *Artificial Intelligence*, Vol. 127, No. 2 (2001), pp. 221-259
- [39] Ghidini, C., and Serafini, L., A Context-Based Logic for Distributed Knowledge Representation and Reasoning, In *CONTEXT*, pp. 159-172, 1999
- [40] Ghidini, C., and Serafini, L., Mapping Properties of Heterogeneous Ontologies, In *1st International Workshop on Modular Ontologies (WoMo 2006), co-located with ISWC*, 2006
- [41] Ghilardi, S., Lutz, C., and Wolter, F., Did I Damage my Ontology? A Case for Conservative Extensions in Description Logics, In *KR*, pp. 187-197, AAAI Press, 2006

- [42] Giunchiglia, F., and Ghidini, C. Local Models Semantics, or Contextual Reasoning = Locality + Compatibility, In *KR*, pp. 282-291, 1998
- [43] Goldszmidt, M., and Pearl, J., On the Consistency of Defeasible Databases, *Artificial Intelligence*, Vol. 52, No. 2 (1991), pp. 121-149
- [44] Guha, R.V., and McCarthy, J., Varieties of contexts, In *CONTEXT*, pp. 164-177, 2003
- [45] Guha, R.V., McCool, R., and Fikes, R., Contexts for the Semantic Web, In Sheila A. McIlraith, Dimitris Plexousakis, and Frank van Harmelen, editors, *International Semantic Web Conference, Lecture Notes in Computer Science*, Vol. 3298, pp. 32-46, Springer-Verlag, 2004
- [46] Haarslev, V., and Möller, R., Racer System Description, In *IJCAR*, pp. 701-706, 2001
- [47] Heinsohn, J., Probabilistic Description Logics, In de Mántaras and Poole [60], pp. 311-318
- [48] Hölldobler, S., Störr, H.-P., and Tran, D.K., The fuzzy Description Logic \mathcal{ALC}_{FH} with Hedge Algebras as Concept Modifiers, *Journal of Advanced Computational Intelligence and Intelligent Informatics*, Vol. 7, No. 3 (2003), pp. 294-305
- [49] Hollunder, B., An Alternative Proof Method for Possibilistic Logic and its Application to Terminological Logics, In de Mántaras and Poole [60], pp. 327-335
- [50] Horrocks, I., Patel-Schneider, P.F., and van Harmelen, F., From *SHIQ* and RDF to OWL: The Making of a Web Ontology Language, *Journal of Web Semantics*, Vol. 1, No. 1 (2003), pp. 7-26
- [51] Horrocks, I., and Sattler, U., A Tableau Decision Procedure for *SHOIQ*, *Journal of Automated Reasoning*, Vol. 39, No. 3 (2007), pp. 249-276
- [52] Horrocks, I., Sattler, U., and Tobies, S., Practical Reasoning for Expressive Description Logics, In *LPAR*, pp. 161-180, 1999
- [53] Jaeger, M., Probabilistic Reasoning in Terminological Logics, In *KR*, pp. 305-316, 1994

- [54] Jaeger, M., Probabilistic Role Models and the Guarded Fragment, *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems*, Vol. 14, No. 1 (2006), pp. 43-60
- [55] Koller, D., Levy, A.Y., and Pfeffer, A., P-Classic: A tractable Probabilistic Description Logic, In *AAAI/IAAI*, pp. 390-397, 1997
- [56] Kutz, O., Lutz, C., Wolter, F., and Zakharyashev, M., \mathcal{E} -Connections of Abstract Description Systems, *Artificial Intelligence*, Vol. 156, No. 1 (2004), pp. 1-73
- [57] Lehmann, D.J., Another Perspective on Default Reasoning, *Annals of Mathematics and Artificial Intelligence*, Vol. 15, No. 1 (1995), pp. 61-82
- [58] Levesque, H.J., and Brachman, R.J., Expressiveness and Tractability in Knowledge Representation and Reasoning, *Computational Intelligence*, Vol. 3 (1987), pp. 78-93
- [59] Lifschitz, V., Minimal Belief and Negation as Failure, *Artificial Intelligence*, Vol. 70, No. 1-2 (1994), pp. 53-72
- [60] López de Mántaras, R., and Poole, D., editors, *UAI '94: Proceedings of the Tenth Annual Conference on Uncertainty in Artificial Intelligence, July 29-31, 1994, Seattle, Washington, USA*, Morgan Kaufmann, 1994
- [61] Lukasiewicz, T., Probabilistic Deduction With Conditional Constraints over Basic Events, In Anthony G. Cohn, Lenhart Schubert, and Stuart C. Shapiro, editors, *KR'98: Principles of Knowledge Representation and Reasoning*, pp. 380-391. Morgan Kaufmann, San Francisco, California, 1998
- [62] Lukasiewicz, T., Probabilistic Logic Programming Under Inheritance With Overriding, In Jack S. Breese and Daphne Koller, editors, *UAI*, pp. 329-336, Morgan Kaufmann, 2001
- [63] Lukasiewicz, T., Probabilistic Default Reasoning With Conditional Constraints, *Annals of Mathematics and Artificial Intelligence*, Vol. 34, No. 1-3 (2002), pp. 35-88
- [64] Lukasiewicz, T., Expressive Probabilistic Description Logics, *Artificial Intelligence*, Vol. 172, Vol. 6-7 (2008), pp. 852-883

- [65] Lutz, C., Walther, D., and Wolter, F., Conservative Extensions in Expressive Description Logics, In *IJCAI*, pp. 453-458, 2007
- [66] McIlraith, S.A., Plexousakis, D., and van Harmelen, F., editors, *The Semantic Web - ISWC 2004: Third International Semantic Web Conference, Hiroshima, Japan, November 7-11, 2004. Proceedings, Lecture Notes in Computer Science*, Vol. 3298, Springer-Verlag, 2004
- [67] McCarthy, J., Circumscription - a Form of Non-monotonic Reasoning, *Artificial Intelligence*, Vol. 13, No. 1-2 (1980), pp. 27-39
- [68] McCarthy, J.L., Generality in Artificial Intelligence, *Communications of the ACM*, Vol. 30, No. 12 (1987), pp. 1029-1035
- [69] McCarthy, J., Notes on Formalizing Context, In *IJCAI*, pp. 555-562, 1993
- [70] McDermott, D.V., and Doyle, J., Non-Monotonic Logic I, *Artificial Intelligence*, Vol. 13, No. 1-2 (1980), pp. 41-72
- [71] Nebel, B., *Reasoning and Revision in Hybrid Representation Systems, Lecture Notes in Computer Science*, Vol. 422, Springer-Verlag, 1990
- [72] Nottelmann, H., and Fuhr, N., Adding Probabilities and Rules to OWL Lite Subsets Based on Probabilistic Datalog, *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems*, Vol. 14, No. 1 (2006), pp. 17-42
- [73] Pan, J., Serafini, L., and Zhao, Y., Semantic Import: An Approach for Partial Ontology Reuse, In *1st International Workshop on Modular Ontologies (WoMo 2006), co-located with ISWC*, 2006
- [74] Paslaru Bontas, E., Context Representation and Usage for the Semantic Web: A State of the Art, Technical Report B-04-30, Freie Universität Berlin, <http://www.inf.fu-berlin.de/inst/pubs/tr-b-04-30.abstract.html>, 2004

- [75] Patel-Schneider, P.F., A Four-Valued Semantics for Terminological Logics, *Artificial Intelligence*, Vol. 38, No. 3 (1989), pp. 319-351
- [76] Patel-Schneider, P.F., and Horrocks, I., DLP and FACT, In TABLEAUX, pp. 19-23, 1999
- [77] Pool, M., and Aikin, J., Keeper and Protégé: An Elicitation Environment for Bayesian Inference Tools, In *Proceedings of the Workshop on Protégé and Reasoning held at the 7th International Protégé Conference*, 2004
- [78] Reiter, R., A Logic for Default Reasoning, *Artificial Intelligence*, Vol. 13, No. 1-2 (1980), pp. 81-132
- [79] Reiter, R., What Should a Database Know?, *Journal of Logic Programming*, Vol. 14, No. 1-2 (1992), pp. 127-153
- [80] Rousset, M.-C., Small Can be Beautiful in the Semantic Web, In McIlraith et al. [66], pp. 6-16
- [81] Serafini, L., Borgida, A., and Tamin, A., Aspects of Distributed and Modular Ontology Reasoning, In *IJCAI*, pp. 570-575, 2005
- [82] Serafini, L., and Tamin, A., Local Tableaux for Reasoning in Distributed Description Logics, In *Description Logics Workshop 2004, CEUR-WS Vol 104*, 2004
- [83] Serafini, L., and Tamin, A., Drago: Distributed Reasoning Architecture for the Semantic Web, In *ESWC*, pp. 361-376, 2005
- [84] Schild, K., A correspondence Theory for Terminological Logics: Preliminary Report, In *In Proc. of IJCAI-91*, pp. 466-471, 1991
- [85] Schmidt-Schauß, M., and Smolka, G., Attributive Concept Descriptions With Complements, *Artificial Intelligence*, Vol. 48, No. 1 (1991), pp. 1-26
- [86] Schreiber, G., and Dean, M., OWL Web Ontology Language Reference, <http://www.w3.org/TR/2004/REC-owl-ref-20040210/>, 2004

- [87] Sirin, E., and Parsia, B., Pellet: An OWL DL Reasoner, In *Description Logics Workshop*, 2004
- [88] Stoermer, H., Introducing Context into Semantic Web Knowledge Bases, In *Proceedings of the CAISE*06 Doctoral Consortium*, 2006
- [89] Stoermer, H., Palmisano, I., Redavid, D., Iannone, L., Bouquet, P., and Semeraro, G., Contextualization of an RDF Knowledge Base in the VIKEF Project, In *ICADL*, pp. 101-110, 2006
- [90] Straccia, U., A sequent Calculus for Reasoning in Four-Valued Description Logics, In Didier Galmiche, editor, *TABLEAUX, Lecture Notes in Computer Science*, Vol. 1227, pp. 343-357, Springer-Verlag, 1997
- [91] Straccia, U., A Fuzzy Description Logic, In *AAAI/IAAI*, pp. 594-599, 1998
- [92] Straccia, U., Reasoning Within Fuzzy Description Logics, *Journal of Artificial Intelligence Research*, Vol. 14 (2001), pp. 137-166
- [93] Straccia, U., Description Logics Over Lattices, *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems*, Vol. 14, No. 1 (2006), pp. 1-16
- [94] Stuckenschmidt, H., and Klein, M.C.A., Integrity and Change in Modular Ontologies, In *IJCAI*, pp. 900-908, 2003
- [95] Tobies, S., Complexity Results and Practical Algorithms for Logics in Knowledge Representation, Ph.D. Dissertation, LuFG Theoretical Computer Science, RWTH-Aachen, Germany, 2001
- [96] Tresp, C., and Molitor, R., A Description Logic for Vague Knowledge, In *ECAI*, pp. 361-365, 1998
- [97] Voutsadakis, G., Slutzki, S., Honavar, V., and Bao, J., Federated *ALCI*: Preliminary report, In *Web Intelligence*, pp. 575-578, IEEE, 2008

- [98] Voutsadakis, G., Slutzki, G., Bao, J., and Honavar, V., *F- \mathcal{ALCI} : A Fully Contextualized, Federated Logic for the Semantic Web*, Technical report, Iowa State University, 2008
- [99] Voutsadakis, G., Slutzki, G., Bao, J., and Honavar, V., *A Distributed Tableau Algorithm for F- \mathcal{ALCI}* , *Preprint*, 2009
- [100] Voutsadakis, G., Slutzki, G., and Honavar, V., *Reasoning with F- \mathcal{ALCI} Over Lattices*, Technical report, Iowa State University, 2009
- [101] Wolter, F., and Zakharyashev, M., *Multi-Dimensional Description Logics*, In Thomas Dean, editor, *IJCAI*, pp. 104-109, Morgan Kaufmann, 1999
- [102] Yang, Y., and Calmet, J., *Ontobayes: An Ontology-Driven Uncertainty Model*, In *CIMCA/IAWTIC*, pp. 457-463, IEEE Computer Society, 2005
- [103] Yen, J., *Generalizing Term Subsumption Languages to Fuzzy Logic*, In *IJCAI*, pp. 472-477, 1991
- [104] Zhao, Y., Wang, K., Topor, R.W., Pan, J.Z., and Giunchiglia, F., *Semantic Cooperation and Knowledge Reuse by Using Autonomous Ontologies*, In *ISWC/ASWC, Lecture Notes in Computer Science*, Vol. 4825, pp. 666-679, Springer-Verlag, 2007